



TAMPEREEN TEKNILLINEN YLIOPISTO

AKI HALME
HIPGUARD – JÄRJESTELMÄN KESKUSYKSIKKÖ

Diplomityö

Tarkastaja: Professori Jukka Vanhala

Tarkastaja ja aihe hyväksytty
tieto- ja sähkötekniikan tiedekuntaneuvoston
kokouksessa 13. elokuuta 2008

TIIVISTELMÄ

TAMPEREEN TEKNILLINEN YLIOPISTO

Signaalinkäsittelyn ja tietoliikennetekniikan koulutusohjelma

HALME, AKI: HipGuard -järjestelmän keskusyksikkö

Diplomityö, 69 sivua, 9 liitesivua

Tarkastaja: Professori Jukka Vanhala

Kesäkuu 2010

Avainsanat: terveydenhuolto, puettava teknologia, liikkeentunnistus, langaton tiedonsiirto, käyttöliittymä

Nykyinen teknologian kehittyminen mahdollistaa soveltuvin osin terveydenhuollon siirtymisen kotiolosuhteisiin, jolloin tarvitaan uusia keinoja ja järjestelmiä hoidon tueksi. Tällä hetkellä käytetyimpiä terveydenhuollon uusia sovelluksia ovat erityyppiset elintoimintojen aktiivisuutta mittaavat järjestelmät kuten sykemittarit. Sen sijaan reaaliaikaisia, käyttäjän raajojen asentoa mittaavia järjestelmiä on tarjolla varsin rajoitetusti.

Tutkimuksen tarkoituksena oli tutkia menetelmiä ja tekniikoita, joilla voidaan toteuttaa järjestelmä henkilön raajan asennon tulkintaan. Diplomityön pääpainona oli suunnitella ja toteuttaa kyseiseen järjestelmään keskusyksikkö, joka on puettava, akkukäyttöinen ja soveltuva pitkäaikaiskäyttöön. Tekniseltä kannalta keskusyksikön tulee tarjota rajapinnat kahdelle eri radiotekniikalle ja kyetä kohtuullisen monimutkaiseen raajan asennon laskentaan. Lopulta tullaan päätelmään onko yleensä mahdollista toteuttaa järjestelmä kohtuullisella panostuksella.

Projektin tuloksena syntyi HipGuard -järjestelmä lonkkaleikkausoperaatiosta toipuville henkilöille. Järjestelmän puettava osa koostuu keskusyksiköstä ja seitsemästä erityisiin housuihin sijoitetusta asentoanturiyksiköstä. Keskusyksikkö tarjoaa rajapinnan kahdelle langattomalle anturitekniikalle, ANT:lle ja Bluetoothille, joiden toimintaa ohjaa 8 -bittinen mikrokontrolleri. Koska kyseessä on puettava järjestelmä, on se pyritty rakentamaan mahdollisimman vähän käyttäjää häiritseväksi.

Lopuksi järjestelmälle suoritettiin niin käyttäjäkeskeisiä kuin teknisiäkin testejä todentamaan toiminta käytännössä. Käyttäjät testit keskittyivät pääosin käytettävyyssnäkökulmiin ja tekninen testaus arvioi ja mittasi muun muassa järjestelmän virrankulutusta ja radiolinkkien suorituskykyä. Myös elektroniikan suojauksen tarve oli keskeinen pohdinnan osa -alue.

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY

Master's Degree Programme in Signal Processing and Communications

HALME, AKI: Central Unit of HipGuard - system

Master of Science Thesis, 69 pages, 9 appendix pages

Examiner: prof. Jukka Vanhala

June 2010

Keywords: healthcare, wearable technology, motion tracking, wireless communication, user interface

Since healthcare is nowadays expanding to home conditions, new kinds of concepts are required to support the treatment at home. Most used healthcare applications, suitable for home treatment at the moment are vital signs monitors (eg. heart rate) with option to call for help in a case of emergency. Instead, real-time applications for tracking person's posture, especially lower limbs, are fairly rare.

The purpose of this research was to investigate methods and techniques that can be used for producing a suitable system for posture recognition. Emphasis of this Thesis was to design and implement a main unit for a system, which is wearable, battery powered and suitable for long-term usage. In technical aspect, the main unit must provide support for two wireless radio technologies and be capable of solving quite complex posture and movement calculations. One final decision to make is whether it is possible to implement such system with reasonable effort.

As a result a wearable HipGuard measurement system for patients recovering from a hip replacement operation was devised. The wearable part of the system consists of a central unit and seven posture sensing units which are located in carefully selected places in a specially made pair of pants. The central unit provides interface for Bluetooth- and ANT-radio technologies and all the computing is handled by an 8-bit RISC microcontroller. Because the system is wearable, the central unit was designed to be as unobtrusive as possible in physical aspect.

Finally some user-based and technical tests were made to confirm the unit's functionality in practice. User-based testing consists of various tests concerning the wearability aspects of the placement of the modules and the physical structure. Technical tests measure the system's power consumption and the performance of the radio links. Also the needs for protecting the electronics have been considered.

ALKUSANAT

Tämä diplomityö on tehty Tampereen Teknillisen Yliopiston Elektroniikan laitoksella Kankaanpään yksikössä. Työn tekemisen mahdollisti TEKES -pohjainen projekti, jossa oli mukana sekä yrityksiä että tutkimuslaitosapuolia. Suuri kiitos kuuluu siis projektiin osallistuneille yrityksille jo rahoituksenkin puolesta. Kiitos myös työni tarkastajalle professori Jukka Vanhalalle, joka kiireisestä aikataulusta huolimatta vastasi työni ohjaamisesta ja antoi arvokkaita neuvoja työn edetessä. Kiitos loistavasta työilmapiiristä kuuluu luonnollisesti koko Kankaanpään Yksikön työkavereille, jotka kokeneempina tutkijoita antoivat erinomaista kritiikkiä ja neuvoja työn edetessä.

Vuosien opiskelun jälkeen jo lapsena alkanut kiinnostus teknisiin laitteisiin kiteytyy tämän diplomityön ja diplomi -insinöörin tutkinnon myötä. Kiitokset siis kotijoukoille mahdollisuudesta ja tuesta toteuttaa tätä kiinnostusta tekniikkaan monissa muodoissa ja erityisesti kiitos Kaisalle, joka jaksoi uskoa tämän työn valmistumiseen.

Kankaanpäässä 30.5.2010

Aki Halme
Jämijärventie 545
38800 Jämijärvi

SISÄLLYS

Tiivistelmä	ii
Abstract	iii
Alkusanat	iv
Termit ja määritelmät	vii
1. Johdanto	1
2. HipGuard – järjestelmän konsepti	3
2.1. Puettava elektronikka teknologiana	3
2.2. Olemassa olevat arkkitehtuurit ja järjestelmät	5
2.3. Toteutettava järjestelmä	7
2.4. Vaatimuksia järjestelmän keskusyksikölle	9
3. Raajan asennon laskemisen teoreettinen tausta	11
3.1. Anturityypit	11
3.2. Algoritmit	14
3.3. Käyttäjän mittaaminen	15
4. Keskusyksikön elektronikan suunnittelu	18
4.1. Käytetyt ohjelmistot	18
4.2. Piirikaaviot ja komponenttivalinnat	18
4.3. Piirilevysuunnittelu ja periaatteet	23
4.3.1. Erityishuomioita taipuisille piirilevyille	24
4.4. Valmiit piirilevysuunnitelmat	26
5. Keskusyksikön toteutus ja integrointi	28
5.1. Piirilevy liitosalustana	28
5.1.1. Piirilevymateriaalit	28
5.2. Piirilevyvalmistus	29
5.3. Kokoonpano	31
6. Keskusyksikön ohjelmisto	35
6.1. Ohjelmiston suunnittelu	35
6.1.1. Ohjelmisto muuhun järjestelmään liittynnän kannalta	35
6.1.2. Ohjelmisto käyttöliittymän kannalta	36
6.2. Kehysformaatti ja tiedonsiirtoprotokollat	37
6.2.1. Anturiverkon ja keskusyksikön välinen yhteyskäytäntö	38
6.2.2. Älypuhelimien ja keskusyksikön välinen yhteyskäytäntö	41
6.3. Ohjelmiston toteutus ja toimintaperiaate	43
7. Kotelointi ja kiinnitys asusteeseen	48
7.1. Puettavuus	48
7.2. Anturit ja keskusyksikkö asusteessa	49
7.3. Joustavat kotelointimateriaalit	50
8. Järjestelmän testaus ja tulokset	54
8.1. Käyttäjäkokeumusperustainen testaus	55
8.2. Tekninen testaus	56

8.2.1.	Ohjelmiston tekninen testaus.....	56
8.2.2.	Virrankulutuksen mittaus.....	60
9.	Yhteenveto	64
	Lähteet.....	66
	Liitteet	70

TERMIT JA MÄÄRITELMÄT

0x[0...F]	Merkintätapa heksadesimaalimuotoiselle datalle.
AFH	Adaptive frequency hopping. Häiriötaajuuksia havaitseva kaistanvaihtelu radiotekniikassa.
ANT	2.4 GHz:n ISM kaistaa hyödyntävä langaton verkkoprotokolla.
BAN	Body Area Network. Käyttäjän kehonläheinen langaton tietoverkko.
bps	bits per second, tiedonsiirtonopeuden yksikkö bittinä/sekunti
broadcast	Yhteislähetys, joka tarkoittaa datavirran tai – paketin lähettämistä ennalta määräämättömälle vastaanottajalle.
BCN	Body Centric Network, käyttäjäkeskeisemmäksi parannettu versio PAN verkosta.
BT	Bluetooth, 2,4GHz lähiverkkostandardi.
DSSS	Direct Sequence Spread Spectrum eli suorasekventointi on toinen CDMA:n toteutustekniikoista.
FH-CDMA	Frequency Hopping Code Division Multiple Access, koodausmenetelmä.
FHSS	Frequency Hopping Spread Spectrum, taajuushyppelyn toteuttava CDMA:n toteutustekniikka.
FR-4	Flame Retardant 4, lasikuitupohjainen ja laajalti käytetty piirilevymateriaali.
FSK	Frequency Shift Keying, taajuusmodulointi.
haptinen	Tuntoaistiin perustuva havainto.
HCI	Host Controller Interface, tietoliikennerajapinta.

ISM-kaista	Industrial, Scientific and Medical on maailmanlaajuinen vapaasti käytettävissä oleva radiotaajuuskaista.
iWRAP	Bluegiga Technologies:n kehittämä ASCII-merkkeihin perustuva rajapinta Bluetooth -moduulin hallintaan.
JEP	juotteenestopinnoite
L2CAP	Logical Link Controller and Adaptation Protocol, Bluetooth-protokolla.
LMP	Link Manager Protocol, Bluetooth-protokolla.
little-endian	Datan tavujärjestys, jossa datan vähemmän merkitsevät tavut talletetaan alempiin muistiosoitteisiin.
MEMS	Microelectromechanical system. Mikroelektromekaaninen järjestelmä.
RMS	Root Mean Square, neliöllinen keskiarvoistusmenetelmä muun muassa jännitteelle.
RS-232	Recommended Standard 232, sarjaliikenneliityntä.
RSSI	Received Signal Strenght Indication. Radiosignaalin tehonsäätelyn mahdollistava havainnointi.
SPI	Serial Peripheral Interface, sarjamuotoinen oheislaiteliityntä
SPP	Serial Port Profile, RFCOMM:n alainen Bluetooth -profiili, jolla voidaan korvata langallinen RS-232 liityntä.
UI	User Interface, käyttöliittymä.
USART	Universal Synchronous/Asynchronous Receiver Transmitter. Yleiskäyttöinen sarjaliikenteen lähetyks ja vastaanottopiiri.
WPAN	Wireless Personal Area Network, lyhyen kantaman langaton tiedonsiirtoverkko.

1. JOHDANTO

Erityisesti nykypäivän terveydenhuollon resurssipula asettaa uusia vaatimuksia käytettävälle hoitomenetelmille. Väestörakenteen ikääntyminen on ollut jatkuvaa vuosikymmenien ajan ja alati tapahtuva sosiaalisten olojen kehittyminen vaatii korkeatasoisia, käyttäjäystävällisiä hoitomenetelmiä. Toisaalta myös koko yhteisön kannalta on eduksi tuottaa yhä tehokkaampia menetelmiä terveydenhuoltoon, jolloin muun muassa hoidon kustannukset saadaan pysymään kurissa. [1]

Viime vuosina onkin markkinoille ilmestynyt monia kaupallisia sovelluksia henkilöiden terveydentilan tarkkailuun. Puettavat ja ”exosensor” -tyyppiset ratkaisut lienevät tällä hetkellä suosituinta tekniikkaa, mutta jo lähitulevaisuudessa uudet nanoteknologian saavutukset tuovat huomattavan paljon lisää mahdollisuuksia. Valtaosa tällä hetkellä saatavilla olevista järjestelmistä keskittyy tavalla tai toisella joko henkilön fyysisen aktiivisuuden mittaamiseen tai kriittisten elintoimintojen, kuten verenpaineen, sykkeen tai EKG:n, tarkkailuun. Näin ollen valmiiden seurantajärjestelmien toiminnot ja mahdollisuudet ovat varsin spesifisiä ja hankalia muokata muihin sovelluksiin sopiviksi. [2]

HipGuard -järjestelmä on kehitteillä erityisesti lonkkaleikkauksesta toipuvien potilaiden kotona suoritettavaa jälkihoitoa silmällä pitäen. Jotta leikkauksesta voidaan toipua tehokkaasti, edellyttää se jalan/lonkan riittävää kuntoutusta, joka esimerkiksi sairaalaolosuhteissa saattaa tulla huomattavan kalliiksi. Toisaalta kuntoutus on vaativa prosessi, jossa raajaa tulee rasittaa aina asiantuntijan määrittämille äärirajoille asti kuitenkin ylittämättä niitä. Kotioloissa, ilman hoitoa tukevaa järjestelmää, on virhesuoritusten todennäköisyys kuntoutuksessa suuri ja seuraukset voivat olla tuhoisia. Juuri tähän ongelmaan etsitään ratkaisua HipGuard -järjestelmällä. Järjestelmän päällepuettava osio koostuu toimintaa ohjaavasta keskusyksiköstä sekä yhteensä 7 asusteeseen sijoitettavasta anturisolmusta. Koko järjestelmäkonsepti sisältää myös kokonaisvaltaisen palvelun mahdollistavia osa -alueita päätyn aina terveydenhuollon ammattilaisille tarjottavaan palvelinratkaisuun. Järjestelmän konseptointi on suoritettu yhdessä Satakunnan sairaanhoitopiirin kanssa, lisäksi mukana on eri tekniikan alojen tutkimusyksiköitä Tampereen Teknillisen Yliopiston piiristä. Päävastuu projektista on Elektroniikan laitoksen Kankaanpään yksiköllä, jonka tutkimustyö fokuksituu jo entuudestaan Puettavaan teknologiaan. Itse järjestelmä on niin ikään toteutettu yhteistyössä yksiköiden kesken.

Tämä diplomityö on jaettu kymmeneen lukuun. Luvut 2 ja 3 tutustuttavat lukijan yleisesti puettavaan teknologiaan sekä esittävät perusteita toteutettavalle konseptille. Lisäksi perehdytään hieman tarkemmin itse laitteen keskusyksikön vaatimuksiin ja toiminnallisuuteen.

Luvussa 4 käydään lyhyesti läpi teoreettista taustaa asennon tulkintaan sekä esitellään muutamia matemaattisia peruseriaatteita raajojen asennon laskemiseen. Samalla esitellään tällä hetkellä yleisimmin asennon tulkintaan käytettäviä elektronisia anturointikomponentteja. Lopuksi perehdytään itse käyttäjän mittaamiseen ja siinä esiin tulleisiin käytännön ongelmiin.

Luku 5 käsittää järjestelmän keskusyksikön elektroniikan suunnittelun komponenttivalintoineen. Esitellään lyhyesti suunnitteluun käytetyt ohjelmistot ja tutustutetaan lukija tietokoneavusteisen piirilevysuunnittelun peruseriaatteisiin unohtamatta joustavien piirilevymateriaalien erityisvaatimuksia. Lopuksi esitellään valmiit piirilevysuunnitelmat, joilla keskusyksikkö tullaan toteuttamaan.

Luvussa 6 on esitelty keskusyksikön varsinainen fyysinen toteutus. Esitellään perinteiset FR-4 -pohjaiset ja joustavat piirilevyt materiaaleina sekä näille sopivia valmistustekniikoita. Kappaleessa tärkeän osakokonaisuuden muodostaa keskusyksikön modulaarisesta rakenteesta aiheutuva integrointi fyysisellä tasolla.

Sulautetussa järjestelmässä välttämätöntä ohjelmistoa ja sen kehitystä käsitellään luvussa 7. Alussa esitellään käytetty kehitysympäristö toimintoineen sekä käydään läpi ohjelmiston vaatimuksia eri osa -alueiden, kuten anturiverkon yhteydenpidon, kannalta. Toinen laajempi kokonaisuus esittelee varsin yksityiskohtaisesti erityisesti järjestelmää varten kehitetyn kehysformaatin ja tavat, joilla sitä hyödynnetään sovelluksessamme.

Luku 8 käsittelee toteutetun elektroniikan suojaamista mekaanisilta rasituksilta, kosteudelta sekä muilta järjestelmän ulkopuolelta laitteistoon kohdistuvilta rasitteilta. Käsitellään myös koteloinnin vaikutusta järjestelmän puettavuuteen ja esitellään tärkeimmät piirilevyjen suojaukseen tarjolla olevat (joustavat) materiaalit.

Luku 9 käsittelee toteutetun järjestelmän testausta niin käytännön olosuhteissa kuin teknisen testauksen kannalta. Luvun alussa luodaan pieni katsaus yleisesti käytössä oleviin sulautetun järjestelmän testausmenetelmiin. Luku sisältää myös testaustuloksia ja esittelee testauksen aikana tehtyjä havaintoja toteutetun keskusyksikön ja järjestelmän toiminnasta ja mahdollisista puutteista.

Lopuksi luvussa 10 summataan projektin kulkua sekä pohditaan sekä keskusyksikön että järjestelmän onnistuneisuutta suhteutettuna asetettuihin tutkimustavoitteisiin.

2. HIPGUARD – JÄRJESTELMÄN KONSEPTI

HipGuard -järjestelmän kokonaiskonseptin esittely on olennaista varsinaisen keskusyksikön toiminnan ja toteutettavien ominaisuuksien ymmärtämiseksi. Konseptissa yhdistyvät myös termit ja käsitteet varsin monelta eri osa -alueelta, joten niiden ymmärtämiseksi on hyvä käydä lävitse muutamia pääkohtia.

2.1. Puettava elektroniikka teknologiana

Puettava teknologia on laaja käsite, joka voidaan jaotella useisiin eri alikategorioihin. Useimmille ensimmäinen puettavuudesta mieleen tuleva teknologia lienee tekstiili- ja kuitu -teknologia, mutta on todellisuudessa paljon muutakin. Puettavassa teknologiassa yhdistyvät nykypäivänä parhaimmillaan useiden, perinteisesti erillään olevien, toimialojen erikoisosaaminen. Merkittävimpinä mainittakoon vaatetusteknologia, tietotekniikka, pukeutumisen sosiologia ja erityisesti työn kannalta kiinnostava puettava elektroniikka, jota seuraavaksi tarkastellaan hieman tarkemmin.

Puettava elektroniikka (*engl. wearable electronics*) yleisesti tarkoittaa mitä tahansa elektronista laitetta, jota tavalla tai toisella voidaan käyttää ja kuljettaa henkilön päällä. Karkeasti puettava elektroniikka voidaan vielä jaotella älyvaatteisiin ja asuste-elektroniikkaan, jolloin jaotteluperusteena on tapa jolla elektroniset komponentit liittyvät asusteeseen (diskreetti vs. integroitu). Älyvaate terminä ei ole aivan vielä vakiinnuttanut asemaansa ja rinnakkaisia nimityksiä (esim. eVaate) kuuleekin usein käytettävän älykkyyttä sisältävästä asusteesta. Älyvaatteessa mahdollinen elektroniikka on usein integroitu osaksi vaatetta, mutta tavallisesti älyvaatteessa älykkyys ei muodostu yksinomaan käytettävästä elektroniikasta. Älykkyyttä asusteeseen voivat tuoda esimerkiksi tietyllä tavalla reagoivat tekstiilit tai vaikkapa parantunut käytettävyys. Yhteinen piirre kuitenkin kaikille älyvaatteille on joko uusien ominaisuuksien toteuttaminen tai olemassa olevien parantaminen.

Toteutettava HipGuard -järjestelmä yhdistelee soveltuvien osien ominaisuuksia molemmilta puettavan elektroniikan osa -alueilta. Toiminnallisen älykkyyden lisäksi asuste sisältää fyysisesti irrotettavissa olevia elektronisia komponentteja, joiden käyttö on kuitenkin välttämätöntä toimivan laitteen aikaansaamiseksi. Asuste -elektroniikalla on jossain määrin varsinaista älyvaatetta pidempi kehitystausta, joka näkyy jo vuosikymmeninen varrella toteutettujen järjestelmien määrässä. Useat tahot mieltävät ensimmäiseksi puettavaksi tietokoneeksi 1960-luvun alussa esitellyn, suunnilleen sikarilaatikon kokoisen laitteen, jonka tarkoitus oli ennustaa rulettipelin lopputulosta

[3]. Tämä ei kuitenkaan luonut lähtökohtia puettavuuden läpimurrolle, vaan ala vaipui unholaan vuosikymmenniksi. Tänä aikana puettavuus oli vain pienten erityisryhmien intressinä. Uudelleen puettavat tietokoneet alkoivat herättää kiinnostusta toden teolla vasta 1990 -luvun alkupuolella, jolloin tekniikka oli kehittynyt uudelle tasolle muun muassa mikro -ohjaimien myötä. Nopeasta kehityksestä huolimatta visiot ja todellisuus eivät vielä tällöinkään kohdanneet, sillä kuluttajaystävälliset laitteet vaativat elektroniikalta toiminnallisuuden lisäksi pientä kokoa. Tähän pienempään kokoluokkaan on päästy kuluttajatasolla vasta aivan viime vuosina parantuneen integroinnin myötä.

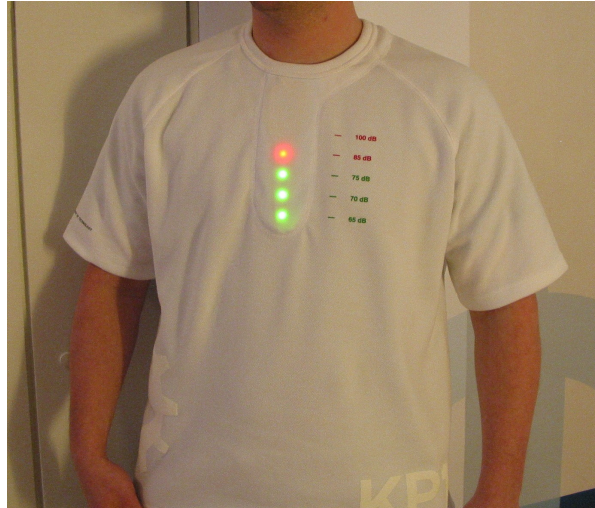
[4]

Älyvaatteen menestyksen määrittää toimintojen lisäksi suurelta osin myös käytettävyys. Käytettävyys on mittari, jolla mitataan tuotteen käytön tehokkuutta, tuottavuutta ja miellyttävyyttä. Loppukäyttäjän tehdessä ostopäätöstä usein mietitään paitsi miltä asuste näyttää ja tuntuu, myös etuja joita mahdollinen hankinta tuo tullessaan. Tällöin tulee myös asusteen käytettävyyden olla kohdallaan joka suhteessa. Mitättömältäkin tuntuva virhe tai epäloogisuus antaa herkästi laitteesta huonon kuvan ja ostopäätös jää tekemättä. Olennaisena älyvaatteen käytettävyyteen vaikuttavana tekijänä on asustemaisuuden lisäksi käyttöliittymä. Tämä käyttäjän ja järjestelmän välinen rajapinta voi sovelluksesta riippuen olla joko kattava, kuten tietokoneen käyttöliittymä, tai minimalistinen, ehkä vain ledistä ja muutamasta napista muodostuva kokonaisuus. Yhteistä kaikille käyttöliittymille on kuitenkin syöte- ja/tai palauteinformaatio, joka voi tapauksesta riippuen olla visuaalista, kuuloon perustuvaa tai haptista eli tuntoaistiin perustuvaa.

Tehonkulutus on niin ikään huomioon otettava tekijä puettavaa järjestelmää suunniteltaessa. Vielä vuosikymmen sitten akkuteknologia, joka on ylivoimaisesti suosituin teholähde puettavassa teknologiassa, ei ollut riittävän kehittynyttä järkevän painoisen laitteen toteuttamiseksi. Eräänlaisena läpimurtona tällä saralla voitaneen pitää Litium -Ion tyyppisten akkujen yleistymistä 2000-luvun alkupuolella. Kyseisen akkutyypin keveys suhteutettuna energiatihyyteen kasvoi nopeasti moninkertaiseksi verrattuna perinteisiin ratkaisuihin. Nykyisin pelkän akun rinnalle on tarjolla useita rinnakkaisia tehonsyöttömenetelmiä aina aurinkokennojärjestelmistä kineettisiin latausjärjestelmiin. Pienitehoisissa järjestelmissä ja suotuisissa olosuhteissa nämä menetelmät saattavat olla jopa yksinään täysin riittäviä tarvittavan tehon aikaansaamiseksi akun toimiessa ainoastaan varalla puskurina. Viimeisimpänä uutuuksena on alan suomalainen osaaja esitellyt jopa taipuisia akkuja. Kustannusten laskiessa nämä akut ovat varteenotettava vaihtoehto puettavassa teknologiassa.

Vähitellen aika alkaa olla kypsä puettaville teknologioille myös sosiologisista näkökulmista. Vielä noin kymmenen vuotta sitten päälle puettava järjestelmä oli useimmiten niin huomiota herättävä, että sen käyttäminen tuntui sosiaalisesti epämieluisalta. Osaltaan tähän sosiaaliseen hyväksyttävyyteen on tuonut helpotusta

yleinen kasvu henkilökohtaisten teknisten laitteiden määrässä ja toisaalta tekniseltä näkökannalta muun muassa anturitekniikoiden miniatyrisointi. Lisäksi on havaittu, että puettavien järjestelmien käyttökynnys madaltuu, kun asuste voidaan pukea ja käyttää normaalin asusteen tavoin. Hyvä esimerkki tällaisesta asusteesta on alla olevan kuvan 2.1 mukainen, Tampereen Teknillisen Yliopiston Kankaanpään yksikön kehittämä Melupaita (*engl. Noise Shirt*).



Kuva 2.1. Melupaita

Melupaita on niin kutsuttu proof-of-concept asuste, jolloin sen tarkoituksena on esitellä tekniikoita ja toteutuksia, joiden tiedetään toimivan ainakin periaatetasolla. Paidan toiminnallisuus on sinänsä yksinkertainen eli se mittaa ympäristön melutasoa ja informoi siitä ledeillä toteutetun asteikon avulla. Toteutuksen varsinainen hienous piilee siinä, että asuste on täysin konepestävä, vieläpä ilman erillisiä toimenpiteitä kuten elektroniikan irrotusta. Käytännössä elektroniikan suojaus niin mekaanista räsytystä kuin kosteuttakin vastaan on saatu aikaan polyuretaanipinnoituksella. Lisäksi tämä akkukäyttöinen järjestelmä latautuu langattomasti roikkuessaan erityisesti asustetta varten kehitetyssä vaateripustimessa.

2.2. Olemassa olevat arkkitehtuurit ja järjestelmät

HipGuard -järjestelmän konseptia suunniteltaessa oli tarkoituksenmukaista tutustua olemassa oleviin järjestelmiin ja niissä käytettyihin arkkitehtuureihin. Samalla perehdyttäisiin käytössä oleviin ratkaisumalleihin erityisesti liikkeenseurantaan (*engl. motion tracking*) hyödyntävissä järjestelmissä ja voitaisiin toisaalta välttää ongelmat, joita muissa järjestelmissä on havaittu. Selvitys tehtiin pääosin sähköistä materiaalia hyödyntäen ja järjestelmien valmistajien luovuttamiin tietoihin perustuen. Tutkimuksessa pyrittiin käyttämään ajankohtaan nähden mahdollisimman tuoretta tietoa, jotta saataisiin selville senhetkinen tekniikan ja toteutusten taso. Selvityksen

avulla voitaisiin perustellusti tehdä päätöksiä omaan HipGuard -järjestelmäämme toteutettavista toiminnoista.

Muutamia järjestelmiä joihin konseptointivaiheessa tutustuttiin:

- *Vivago -hyvinvointiranneke* on turvalaite, jonka tarkoituksena on automaattisesti seurata henkilön hyvinvointia ympäri vuorokauden. Laite omaksuu kalibrointivaiheessa käyttäjänsä aktiivisuustason mikrotason liikkeistä sekä ihon lämpötilasta ja sähkönjohtavuudesta. Muutokset aktiivisuustasossa aiheuttavat automaattisesti hälytyksen seurannasta vastaavalle henkilölle. Vastaavasti käyttäjä voi manuaalisesti kutsua apua laitteen välityksellä. Laitteisto koostuu itse rannekkeesta sekä tukiasemasta, jonka kautta tiedot tapahtumista voidaan välittää normaalin puhelinverkon kautta eteenpäin. [5]
- *MDKeeper* on niin ikään rannelaite henkilön terveydentilan seurantaan. Laite kykenee mittaamaan sydänsähkökäyrää, sykettä sekä veren happisaturaatiota. Tämäkin järjestelmä mahdollistaa kerätyn informaation lähettämisen eteenpäin GSM -verkon välityksellä. Koska järjestelmän kaikki toiminnot on integroitu yhteen laitteeseen, on se fyysiseltä kooltaan epäkäytännöllinen jatkuvaan käyttöön. Järjestelmän tiedetään kuitenkin olevan käytössä muutamissa sairaaloissa. [6]
- *Wealthy – Wearable Health Care System* on järjestelmä, joka osoittautui myöhemmin mielenkiintoisimmaksi omaa projektiamme ajatellen. Tässä tutkimusprojektina toteutetussa järjestelmässä mittaukseen on käytetty hajautetusti tekstiilisiä antureita, jotka yhdistyvät keskusyksikköön. Laitteen tehtävänä on aiempien järjestelmien tapaan kerätä tietoja elintoiminnoista kuten aktiivisuudesta ja ihon lämpötilasta. Omasta suunnitelmasta poiketen, anturien liityntä keskusyksikköön on kuitenkin toteutettu johdoilla. Antureilta saapuvan informaation esiprosessointi keskusyksikössä vähentää GSM -verkon ylitse siirrettävän datan määrää. [7]

Osassa järjestelmiä painotettiin selvästi tekstiilisiä tai puettavia antureita ja toisissa taas pääpaino oli koko konseptissa, joka jollain tavalla mahdollistaisi terveydenhuollon tehostamisen. Samaten käytettyjen antureiden langaton tiedonsiirto keskusyksikölle ja integroitu älykkyys vaihtelivat suuresti järjestelmästä riippuen. Langattomissa anturiratkaisuissa älykkyys oli selvästi tärkeä tekijä, joka mahdollisti pienemmät datansiirtomäärät järjestelmän sisällä.

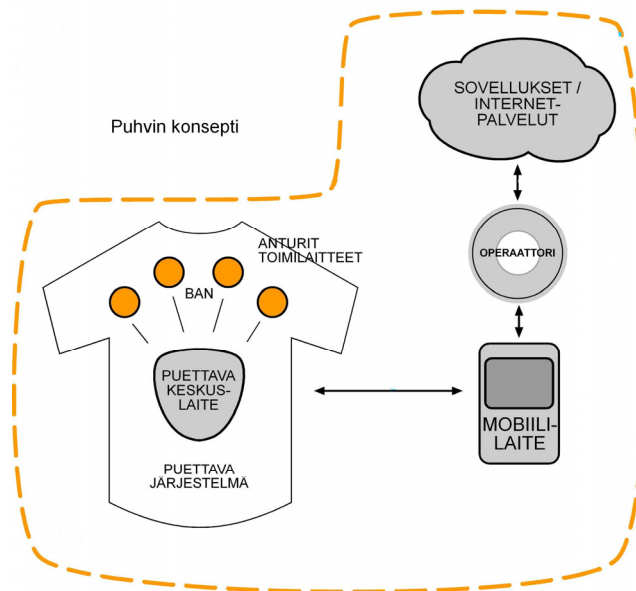
Järjestelmissä oli selvästi havaittavissa myös yhteisiä piirteitä. Ensinnäkään mitään tutkituista järjestelmistä ei ollut täysin integroitu osaksi asustetta. Yleisesti anturielementit oli integroituna asusteessa mutta varsinainen keskusyksikkö oli erillinen

puettava laite, joka sijoitettiin asusteessa olevaan taskuun. Jo tämä antoi viitteitä keskusyksikön integroinnin haasteellisuudesta. Toinen merkittävä yhteinen tekijä on langaton tiedonsiirto järjestelmästä ulospäin mikä onkin luonteva ratkaisu infrastruktuurin ollessa tarjolla. Käytetyimmän GSM/GPRS -ratkaisun myötä järjestelmät ovat käytettävissä lähes missä tahansa. Suurelta osin näiden esitietojen pohjalta määriteltiin Puhvi -projektin konsepti.

2.3. Toteutettava järjestelmä

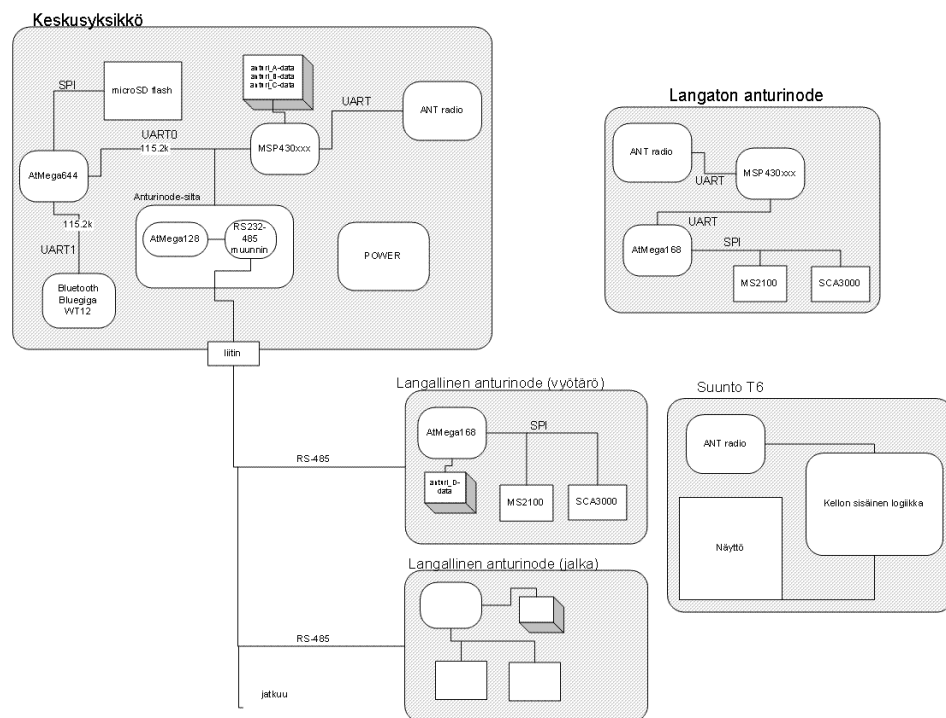
Puettava hyvinvointiteknologia -hankkeessa (lyh. Puhvi) haluttiin tutkimussuunnitelman mukaisesti tutkia puettavan teknologian tuomia mahdollisuuksia terveyden ja hyvinvoinnin hoidossa ja ylläpidossa. Tavoitteena oli terveydenhuoltoalan rajatun osa-alueen toimintojen tukeminen hyödyntämällä erityisesti puettavan teknologian vahvuuksia, kuten käyttäjää kuormittamattomia pitkäaikaismittauksia. Projektin puitteissa tuotettiin useita perusteltuja palvelukonsepteja, joista lopulta toteutettiin yksi esimerkkitapaus toimivaksi prototyypiksi.

Puhvi -hankkeen toteutettavaksi prototyypiksi valittiin lonkkaleikkauspotilaan kotona tapahtuvaa jälkiseurantaa ja kuntoutumista tukeva puettava järjestelmä: HipGuard. Konsepti suunniteltiin palvelun tasolle ja projektin ensimmäisen vaiheen puitteissa asetettiin tavoitteeksi toteuttaa toimiva prototyyppi ainakin henkilökohtaisen elektroniikan osalta, kuitenkin muiden palvelukonseptin osien vaatimukset huomioiden.



Kuva 2.2. Puhvi palvelukonsepti periaatetasolla

Kuvassa 2.2 on nähtävissä konseptisuunnitelma projektin käynnistymisvaiheessa. Projektin edetessä suunnitelmat hioutuivat vaihe vaiheelta tarkemmiksi. Pääpiirteissään järjestelmän henkilökohtainen, puettava osa koostuu asusteeseen kiinnitettävästä tai integroitavasta keskusyksiköstä sekä hajautetusti sijoitelluista anturisolmuista. Konseptikuvasta poiketen lopullinen toteutettu asuste oli kuitenkin housut, eikä paita kuten kuva 2.2 antaa ymmärtää. Keskusyksikön tehtävänä on muuntaa anturisolmuilta saapuvaa mittausdataa paremmin käytettävään muotoon ja lähettää data edelleen langatonta Bluetooth -yhteyttä hyödyntäen sovelluksessa käytettävälle päätelaitteelle. Sovelluksesta riippuen tämä päätelaite voi olla esimerkiksi tavallinen PC varustettuna Bluetooth -ominaisuudella tai esimerkiksi alkuperäisen konseptisuunnitelman mukainen Nokia E60 älypuhelin. Molemmissa tapauksissa päätelaitteille tullaan toteuttamaan asianmukainen ohjelmisto, jolla järjestelmän havaitsemat tapahtumat ovat tarkasteltavissa numeerisessa ja/tai visuaalisessa muodossa. Lisäksi keskusyksikön tehtävänä on toimia käyttöliittymärajapintana käyttäjän ja järjestelmän välillä sekä huolehtia käyttäjälle annettavista varoituksista saapuvan mittausdatan perusteella.



Kuva 2.3. HipGuard – järjestelmän henkilökohtainen osa

Sovelluksessa käytettävät anturisolmut voidaan prototyyppilaitteessa karkeasti jaotella kahteen tyyppiin; langallisiin ja langattomiin. Solmujen sisältöä lohkokaavio tasolla selvittää yllä esitelty kuva 2.3. Erityyppisten antureiden käyttö antaa mahdollisuuden tiedonsiirtomenetelmien vertailuun esimerkiksi luotettavuuden ja datansiirtokapasiteetin kannalta. Lisäksi järjestelmään toteutetaan niin ikään

langattomasti toimiva, kenkään sijoitettava, jalan painorasiasta mittaava anturointi. Painoa mittaavasta pohjallisesta on kirjoitettu projektin aikana julkaisu, joka selvittää kengän anturointiperiaatetta [8]. Anturi itsessään on pohjallisena toteutettava kapasitanssien muutoksiin perustuva yksikkö, jonka mittausdata siirretään langattomasti järjestelmän keskusyksikölle ja siitä edelleen eteenpäin päätelaitteelle.

Järjestelmän konseptista on myös kirjoitettu projektin puitteissa julkaisu, jossa järjestelmän konseptointi on selvitetty perinpohjaisesti. [9]

Kuvasta 2.3 selviää järjestelmän päälle puettava osa ainoastaan lohkokaa viot asolla. Tarkempi analyysi käytetyistä komponenteista ja ratkaisuista esitellään myöhemmässä kappaleessa 5 joka käsittelee Elektroniikan suunnittelua ja komponenttivalintoja.

2.4. Vaatimuksia järjestelmän keskusyksikölle

Järjestelmän ytimen puettavalla osa-alueella muodostaa laitteen keskusyksikkö. Laitteen yksityiskohtaista suunnittelua varten tulee suunnittelijalla olla selvillä vaatimukset joihin laitteen pitää vastata. Hyvä vaatimusmäärittely selkeyttää projektin etenemistä ja usein vaatimusmäärittelyn yhteydessä on hyvä suunnitella tapoja, joilla järjestelmää voidaan ja pitää testata myöhemmässä vaiheessa. Tämä vaatimusten yksityiskohtien muuttuminen projektin loppuvaiheessa realisoituikin ongelmiksi, joiden ratkaiseminen kulutti huomattavasti resursseja kriittisessä vaiheessa. Tästä esimerkkinä mainittakoon tasapainottelu langattomien ja langallisten anturisolmujen käytön välillä. Lopulta oli syytä kyseenalaistaa tarvittiinko langatonta ratkaisumallia tässä sovelluksessa laisinkaan.

Kokonaisjärjestelmään toteutettava monimuotoinen toiminnallisuus asettaa tiukat kriteerit myös keskusyksikön ominaisuuksille. Keskusyksikkö on siis laite, jonka kautta kaikki anturidata sekä verkkoliikenne kytkeytyvät eri osakokonaisuuksien välillä. Käytännössä tämä tarkoittaa toimimista siltana ANT – ja Bluetooth -protokollien välillä. Yksi keskusyksikön haasteista onkin kahden samaa taajuuskaistaa käyttävän radioprotokollan yhteensovittaminen. Toinen merkittävä keskusyksikön tehtävä on huolehtia anturiverkon alustuksista ja ylläpidosta sekä muodostaa anturiverkolle suuntautuvia datapyyntöjä. Lisäksi keskusyksikön tulee tarjota käyttäjälle helppokäyttöinen käyttöliittymä, joka myös kykenee informoimaan käyttäjää älypuhelimien ohjelmiston välityksellä.

Kuten aiemmin mainittiin, asusteen puettavuutta voidaan merkittävästi parantaa suunnittelemalla asusteen elektroniikka mahdollisimman huomaamattomaksi. Tästä johtuen keskusyksikön lopullisesta versiosta on tarkoitus tehdä taipuisa, joten

taipaisuuteen liittyviä näkökohtia tullaan ottamaan huomioon jo ensimmäisiä prototyyppejä suunniteltaessa. Tällöin harppaus lopulliseen versioon saadaan sulavammaksi, kun jo toimiviksi havaittuja komponenttiasetteluja (*engl. layout*) ei tarvitse tehdä uudestaan.

Muita määrittelyn arvoisia tekijöitä, jotka tuli ottaa huomioon:

- Keskusyksikön virrankulutus tulee säilyttää niin pienenä, että pitkäaikaismittaus on mahdollista. Tämän järjestelmän kohdalla pitkäaikaismittauksella tarkoitetaan vuorokausirytmien hereillä oltavaa aikaa. Ihanne toteutettavalle järjestelmälle olisi siis noin 12 -14 tunnin yhtäjaksoinen toiminta -aika.
- Vaatimukset suorituskyvyn suhteen olivat määrittelyvaiheessa melko epäselvät, sillä vastaavaa prosessointia suorittavaa järjestelmää ei aiemmin ollut yleisesti käytössä. Tiedossa oli ainoastaan, suurella anturidatan määrällä asennon laskenta muodostuisi todella raskaaksi, ellei jopa mahdottomaksi. Raskaampaan laskentaan kykenevään 32 -bittiseen suorittimeen ei kuitenkaan haluttu siirtyä suuren tehontarpeen vuoksi.
- Mukana kannettavalle ja päälle puettavalle laitteelle oli luonnollisesti suotavaa pieni koko, mukava muotoilu ja silmää miellyttävä ulkoasu. Prototyyppivaiheessa ulkoasu keskusyksikön jätettiin kuitenkin toisarvoiseen asemaan ja pyrittiin toteuttamaan toiminnallisesti järkevä laitteisto, vaikka ulkoisen olemuksen kustannuksella.

3. RAAJAN ASENNON LASKEMISEN TEOREETTINEN TAUSTA

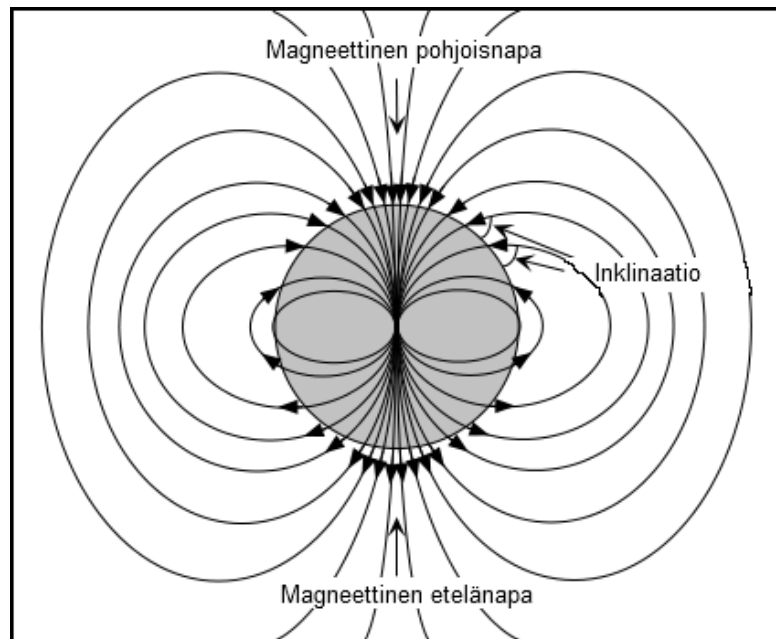
3.1. Anturityypit

Raajan asennon anturointi voidaan yleisesti toteuttaa monella vaihtoehtoisella menetelmällä perustuen esimerkiksi gyroskooppeihin, magneettiantureihin, kiihtyvyysantureihin tai jopa mekaanisiin antureihin. Käytännössä parhaaseen lopputulokseen päästään useimmiten näiden anturityyppien sopivalla yhdistelmällä. Tutustuminen eri anturityyppeihin on ensiarvoisen tärkeää laskennan kannalta, sillä saatava mittausdata muodostaa pohjan kehitettävälle laskentamenetelmille. Tarkastellaan hieman tarkemmin asennontulkintaan yleisimmin käytettäviä sensoreita.

Magneettianturit eli kompassit mittaavat nimensä mukaisesti maan magneettikenttää ja siten anturin asentoa suhteessa tähän kenttään. Perinteisellä 2 -akselisella elektronisella magneettianturilla mitataan ainoastaan maanpinnan tasossa olevia magneettikentän komponentteja. Tällöin saadaan melko suoraviivaisesti laskettua kulma maan magneettisen pohjoisen ja kulkusuunnan välillä. Sinänsä asennonmittaukseen sopivan 2 -akselisen magneettianturin käyttämisestä asennontulkinnassa on kuitenkin monia haittapuolia, joiden vuoksi pelkästään niitä hyödyntäen mitattu asento ei useimmiten ole yksikäsitteinen. Magneettikenttä on vektorisuure, jolloin sille määritellään suuruuden lisäksi suunta. Maan magneettikentän suuruus ja suunta eivät ole ajan ja paikan suhteen vakioita ja siitä syystä magneettikentälle on määriteltä kaks yleistesti käytettyä, paikkaan sidottua, lisämääritettä; inkliinaatio ja deklinaatio. Inkliinaatio määrittelee magneettikentän kaltevuuskulman vaakatasoon nähden ja deklinaatio poikkeaman magneettisen pohjoisnavan ja maantieteellisen pohjoisen välillä. Suomessa deklinaation suuruus vaihtelee 4-10 asteen välillä ja inkliinaatio on noin 73 astetta. Käytännössä näin suuri inkliinaatio tarkoittaa, että magneettikentän suunta poikkeaa ainoastaan noin 17 astetta gravitaatiokentän suunnasta. [10]

Magneettikenttä vaihtelee myös ajan suhteen, jonkin verran jopa tunneittain. Magneettisten myrskyjen ollessa voimakkaimmillaan saattavat nämä muuttaa anturit osoittamaa suuntaa jopa +/- 0.3 asteen verran. Lisäksi magneettianturit ovat hyvin herkkiä muille ulkoisille, luonnosta riippumattomille, häiriötekijöille kuten ympäristön magneettisille signaaleille ja ferromagneettisille materiaaleille. Esimerkiksi suurten sähkövirtojen aiheuttamien magneettikenttien vaikutus tulee pyrkiä kumoamaan kalibroitaessa mittausjärjestelmää käyttöympäristöön. Periaatteellinen kuva maan

magneettikentän kenttäviivoista on nähtävissä kuvasta 3.1. Kyseisessä kuvassa on myös esitetty edellä määritellyn inkliinaatiokulman sijoittuminen suhteessa kenttäviivoihin.



Kuva 3.1. Maan magneettikenttä

Elektronisen kompassin tulee löytää suunta horisontaalitasossa ja jättää huomiotta magneettivuon vertikaali komponentti. Vanhemmilla 2 -akselisilla antureilla tämä on kuitenkin epäkäytännöllistä ja toimintatapa muun muassa estää kompensoinnin mittalaitteen itsensä sisältämän, magneettivuota vääristävän metallin suhteen. Toimenpide olisi kuitenkin tarpeellinen, sillä sensorit eivät ole kiinteitä suhteessa laitekoteloon. Moderneilla 3 -akselisilla magneettiantureilla voidaan edellisistä poiketen laskea koko magneettinen kenttävektori. Kokonaiskenttävektorista voidaan edelleen erottaa horisontaali- ja vertikaalikomponentit käyttäen hyväksi vaikkapa kiihtyvyyssantureilta saatua informaatiota. Kyseistä järjestelyä käyttäen voidaan kompassivirheen suuruus määrittää kaavalla,

$$\text{Compass error} = \tan^{-1} (\sin (\epsilon) \tan (I)) , \quad (3.1)$$

jossa ϵ on kompassin poikkeama pystysuunnasta (*engl. tilt*)
ja I on magneettinen inkliinaatio.

Virheen suuruutta laskettaessa saadaan yllä olevaan kaavaan kompassin poikkeama pystysuunnassa esimerkiksi jäljempänä esiteltävältä 3 -akseliselta kiihtyvyyssanturilta. Vastaavasti paikallinen inkliinaatio on Suomen olosuhteissa keskimäärin aiemmin

mainittu 73 astetta. Mainittakoon, että kaavaa 3.1 käyttäen suuntimavirheeksi Suomen oloissa saadaan 3,3 astetta 1 asteen pystykallistumalla. Kuten aiemmin tuli esille magneettikentän suunta ja suuruus on verrattain nopeasti vaihteleva parametri, joten se aiheuttaa omat ongelmansa laskenta -algoritmiin. [11]

Myös kiihtyvyys on vektorisuure, jolloin sille voidaan määritellä sekä suunta että suuruus. Kiihtyvyysantureilla raajan asentoa voidaan mitata käyttäen hyödyksi maan vetovoiman vaikutusta. Kappaleen asento voidaan teoriassa laskea jo 2 -akselisen kiihtyvyysanturin arvoista, mutta tällöin on suuri riski päätyä asentoon jossa mittaus epäonnistuu. Tällainen tilanne syntyy esimerkiksi silloin kun maan vetovoima on kohtisuorassa molempien kiihtyvyysakselien suhteen. Ongelmasta päästään kuitenkin eroon käyttämällä 3 -akselisia kiihtyvyysantureita, jolloin pystysuoraa kiihtyvyysskomponenttia voidaan käyttää muiden anturityyppien kompensointitietona, kuten jo aiemmin kompassin yhteydessä esitettiin. Eli pelkillä kiihtyvyysantureilla ei nähdä suuntaa painovoimavektorin ympäri. Akseleiden määrän lisäämisestä huolimatta kiihtyvyysantureillakin on ongelmansa, jotka estävät yksikäsitteisen asennon tulkinnan pelkästään yhdellä anturityypillä. Suuri ongelma on myös kiihtyvyysantureiden kyky havaita sekä staattista että dynaamista kiihtyvyyttä. Raajan liikkeestä aiheutuva dynaaminen kiihtyvyys tulisi saada poistettua signaalista, jolloin jäljelle jäisi ainoastaan asennon määrittämiseen vaadittava staattinen kiihtyvyys. Todellisuudessaan varsinaista staattista kiihtyvyyttä ei ole olemassa, mutta koska oma koordinaatistomme on suhteutettu maapalloon, maan keskipistettä kohden vaikuttava kiihtyvyyden komponentti voidaan ajatella staattiseksi. Maapallon vetovoima siis tuottaa asennon tulkintaan tarvittavan staattisen kiihtyvyyden. [12]

Gyroskooppeja käytetään raajan asennon tulkinnassa lähinnä parantamaan asentoanturijärjestelmän dynaamisia ominaisuuksia, jolloin se tekee järjestelmän herkemmäksi pienillekin muutoksille. Gyroskooppi mittaa paikallista kulmakiihtyvyyttä ja kappaleen asento voidaan sen avulla määrittää yksikäsitteisesti. Tällöin kuitenkin vaatimuksena on, että kappaleen alkuasento pitää olla tiedossa. Asennon tulkinta ja liikkeiden seuranta tapahtuu gyroskoopissa integroimalla niiden tuottamia signaaleja. Pääosin tästä johtuen asennonmittausta ei voida käytännössä suorittaa luotettavasti kovin pitkää aikaa ilman järjestelmän kalibrointia. Integroimalla saatu signaali ajautuu mittausepävarmuuden vuoksi koko mittauksen ajan kauemmas todellisesta arvosta. Gyroskoopin selkeästi negatiiviset ominaisuudet, varsinkin puuttavien järjestelmien kannalta, liittyvät anturin fyysiseen kokoon ja virrankulutukseen. Toteutustapana gyroskooppi elektroniikan komponenttina on MEMS -tyyppinen ratkaisu. [13]

Hieman vanhempaa mittausmenetelmää edustavat erilaiset mekaaniset anturit. Esimerkkinä näistä mainittakoon asusteeseen integroitavat venymäliuskat. Mekaanisten antureiden ongelmana voidaan pitää tarvetta kiinnittää käytettävät anturit lähelle mitattavaa niveltä, jolloin ne mittaavat nivelen liikettä ainoastaan yhdessä tasossa.

Lisäksi jokainen mittauksessa mukana oleva nivel on mitattava erikseen, vieläpä usein rakenteeltaan erilaisella anturilla. [14]

3.2. Algoritmit

Nykyaikaisimmat asennon tulkintaan käytettävät algoritmit perustuvat suurimmaksi osaksi unkarilaisen professorin Rudolf E. Kalmanin kehittämään rekursiiviseen tilaestimointialgoritmiin. Tämä niin kutsuttu Kalman -suodatin on hyvin laajalti hyväksytty perusteoria ja suhteellisen pitkän historiansa aikana saavuttanut suuren suosion monilla eri sovellusaloilla. Merkittävimpinä näistä mainittakoon juuri oman sovelluksemme kannalta mielenkiintoinen paikannus- ja asennontulkinta. Tämän lisäksi Kalman -suotimeen perustuvia järjestelmiä tiedetään olevan käytössä ainakin seismisen datan prosessoinnissa, ydinvoimaloiden säätöjärjestelmissä, puheen- ja kuvan tunnistusjärjestelmissä sekä erilaisissa navigointi- ja paikannusjärjestelmissä. [15]

Kalman -suotimen sovellusaloihin tutustumisen jälkeen oli selvää, että kyseinen algoritmi olisi miltei täydellinen omaa sovellustamme ajatellen. Algoritmiin tutustuminen aloitettiin täysin teoreettiselta pohjalta perusperiaatteisiin nojautuen. Suppeasti määriteltynä Kalman -suodatin on rekursiivinen tilaestimointialgoritmi. Tämä algoritmi pyrkii selvittämään dynaamisen lineaarisen stokastisen järjestelmän tilan kohinaa sisältävien mittaustulosten pohjalta. Kriteeri, jolla Kalman -suodin selvittää järjestelmän tilan on ennustevirheen neliösumman minimointi. Kalman -suotimessa järjestelmän tilaesitysdynamiikka oletetaan tunnetuksi muodossa:

$$x(k) = Ax(k-1) + Bu(k) + w(k-1), \quad (3.2)$$

joka tunnetaan dynamiikkayhtälönä sekä

$$z(k) = Hx(k) + v(k), \quad (3.3)$$

mittausyhtälö

Yllä olevan yhtälössä järjestelmän matriisit A , B ja H voivat olla myös ajasta riippuvia. Edelleen Kalman-suotimeen liittyvät perusoletukset ovat

- Satunnaisvektoreiden y :n ja x :n yhteys on lineaarinen ja yhteisjakauma tunnetaan.
- Systeemin prosessi- ja mittaushäiriöt $w(k)$ ja $v(k)$ ovat keskenään riippumattomia nollakeskiarvoista kohinaa. [15]

Suotimen perusperiaatteisiin perehtymisen jälkeen oli projektin osalta selvää, että laskenta -algoritmin sujuva toteutus vaatisi mittaustekniikan huomattavasti

syvällisempää tuntemusta kuin mitä projektin puitteissa oli mahdollista saada. Kalman-suodinta hyödyntäen järjestelmä jättää kuitenkin mielenkiintoisen jatkokehitysmahdollisuuden toisen projektin puitteissa.

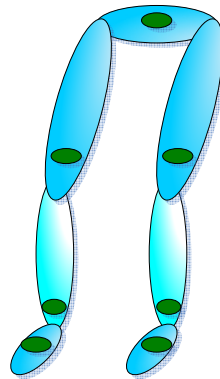
Laitekehityksen ensimmäisessä vaiheessa asennontulkintaan tullaan näin ollen käyttämään huomattavasti yksinkertaisempaa algoritmia. Tähän on monia syitä, joista päälimmäisenä esiin nousee Kalman -suotimen monimutkaisuuden myötä käytettävän suorittimen melko rajallinen laskentakapasiteetti. Erityisesti reaaliaikaisuuteen pyrkivä liukulaskenta on haasteellinen tehtävä käytettävälle prosessortyypille. Mahdollisten jatkoprojektien kannalta onkin olennaista suoritinkapasiteetin lisääminen. Tällöin esimerkiksi prototyypiversion kahdella mikrokontrollerilla toimiva kokonaisuus voitaisiin toteuttaa yhdellä suorittimella. Teoriassa kapasiteettia on mahdollista kasvattaa helpostikin, mutta koska kyseessä on kannettava laite, tulee myös suorittimen virrankulutukseen kiinnittää riittävästi huomiota. Laskentatehoa kasvatettaessa virrankulutus tunnetusti kasvaa, toisaalta tätä on kompensoimassa suorittimien jatkuva kehitys.

3.3. Käyttäjän mittaaminen

Käyttäjän liikettä ja asentoa mitataan tässä sovelluksessa maksimissaan 7 anturisolmulla. Kukin anturisolmu sisältää sekä 3 -akselisen kiihtyvyysanturin että 2 -akselisen magneettianturielementin, jolloin muodostuvalla anturiverkostolla tulisi pystyä kaikissa tilanteissa määrittämään raajan asento yksiselitteisesti. Tällä kokoonpanolla saadaan kiihtyvyysanturin avulla määritettyä raajan liike ja asento kahdessa suunnassa ja kolmas akselisuunta saadaan selville magneettianturin avulla. Koska halutaan mitata ”kiellettyjä” asentoja ovat säären ja reiden magneettianturit vähemmän merkitsevässä asemassa kiihtyvyysantureihin nähden. Henkilön seisoessa eli jalkojen ollessa lähes gravitaation suuntaisesti on magneettiantureiden tuottama data liian vääristynyttä, jotta siitä voitaisiin saada käyttökelpoista. Edes käyttämällä arvojen kompensointiin kiihtyvyysantureiden dataa ei ole mahdollista saada mittausta riittävän luotettavaksi. Näin ollen magneettiantureita voidaan käyttää tässä tilanteessa ainoastaan määrittämään mihin tason neljännekseen jalka on kallellaan.

Vastaavasti raajan asennon ollessa ortogonaalinen eli kohtisuora verrattuna gravitaation suuntaan, voidaan kiertymä suhteessa magneettikenttään sen sijaan määrittää suhteellisen luotettavasti. Tämä ehto toteutuu esimerkiksi henkilön ollessa istuma- tai makuuasennossa. Istuma -asennossa reiden päälle sijoitettu magneettianturi ilmoittaa reiden kulman sivuttaissuunnassa vartalon keskilinjaan nähden, jos vain vertailtava anturipari on sijoitettu vyötärölle. Säärten ja reisien magneettiantureista on siis hyötyä lähinnä silloin kun jalat ovat erisuuntaiset suhteessa gravitaatioon. Kenkiin

sijoitetuilla magneettiantureilla saadaan kuitenkin määritettyä kiertymä pitkien luiden, reisiluun ja sääriluun, suhteen.



Kuva 3.2. Anturien sijoittelu

Kiihtyvyysantureita käytetään mittaamaan reiden ja säären kulmia kahdessa ulottuvuudessa (x - y ja y - z) suhteessa vyötäröön sekä polven kulmaa yhdessä suunnassa (y - z). Periaatteellinen anturien sijoittelu selviää yllä olevasta kuvasta 3.2. Kuvassa anturit sijaitsevat vihreällä värillä merkityillä alueilla ja jokainen anturiyksikkö siis sisältää sekä kiihtyvyys- että magneettianturin. Toiminnan selkeyttämiseksi käsitellään selkeyden vuoksi vain toista raajaa, jolloin asennon määrittämiseksi yksikäsitteisesti vaaditaan 4 anturiyksikköä. Ensimmäinen tapa mitata raajojen asentoa on hyödyntää kiihtyvyysarvoja kompassisuunnan kompensoinnissa, kuten kappaleessa 4.1 esitettiin Anturityyppien yhteydessä. Kutsuttakoon tätä menetelmää absoluuttiseksi menetelmäksi. [16]

Raajojen asennon mittauksen aloittamiseksi tarvitaan niin kutsuttu referenssihetki, jolloin käyttäjä seisoo normaalissa asennossa ja anturiarvot tallennetaan myöhemmin laskennan aikana tapahtuvaa vertailua varten. Referenssihetkeksi valikoitui testien perusteella sopivaksi viiden sekunnin keskiarvo nivelten nollakulmasta. Tämän kalibroinnin tarkoituksena on kumota mahdollisesti ympäristössä vaikuttavien erityisesti magneettikenttää muokkaavien tekijöiden vaikutuksia. Koska kyseessä ovat ”normaalit” päälle puettavat housut, tulee referenssihetken kalibroinnissa osaltaan minimoitua myös antureiden fyysisestä kiertymisestä raajan ympäri johtuvat epätarkkuudet. Juuri anturien kiertymisen estäminen olikin puettavuuden suurimpia haasteita projektissa. [16]

Edellä esitellyn absoluuttisten arvojen laskennan lisäksi toinen mahdollinen tapa asennon laskentaan on vertailla kahden eri anturin muutoksia suhteessa toisiinsa. Jälkimmäisen tavan eduksi havaittiin kyky kompensoida paremmin nopeista liikkeistä aiheutuvia häiriöitä. Suoritettujen testimittausten perusteella tultiin lopputulokseen, että vertaamalla kiihtyvyysarvojen muutosta suoraan alussa mitattuun referenssiarvoon tai

vertaamalla arvoja suoraan toisen anturin arvoihin, ei mittaustarkkuudessa ole olennaista eroa liikkeiden ollessa rauhallisia. Käytännössä tämä tarkoittaa normaalia liikkumista, sillä esimerkiksi juostessa mitatut hetkelliset kiihtyvyydet aiheuttavat huomattavaa virhettä mittauksiin. Referenssitilamenetelmää käyttämällä anturiverkon kuormitusta voidaan tasata, koska mittapisteen (eli anturisolmun) kulma voidaan määrittää jo itse solmussa. Näin jokainen anturisolmu mittaa asentoaan verrattuna nollapisteeseen ja nivelten kulmat voidaan lasketa vertaamalla mittapisteiden kulmia toisiinsa.

4. KESKUSYKSIKÖN ELEKTRONIIKAN SUUNNITTELU

4.1. Käytetyt ohjelmistot

Järjestelmänä toteutettava keskusyksikkö on kokonaan digitaalinen, mistä johtuen suunnittelussa ei ole erityistä tarvetta käyttää analogista simulointiohjelmistoa, esimerkiksi Spice:ä. Orcad -piirisimulaattoria, joka sisältää muun muassa PSpice simulaattorin, käytettiin kuitenkin pienempien osakokonaisuuksien suunnittelun tueksi. Kyseinen ohjelmisto tarjoaisi myös työkalut digitaalisen signaalin laadun arviointiin (*engl. Signal Integrity*), joka siis käsittää pääpiirteissään ylikuulumisen ja signaalin heijastuksien suuruuksien arvioinnin. Tätäkään ei kuitenkaan katsottu tarpeelliseksi, sillä järjestelmässä käytettävät taajuudet ovat suhteellisen pieniä. Sen sijaan piirilevyjen suunnittelussa keskityttiin käytännön kokemuksen perusteella toimiviksi havaittuihin ratkaisuihin.

Tällä hetkellä on laajalti käytössä kaksi erityisesti piirikaavioiden ja -levyjen suunnitteluun tarkoitettua ohjelmistoa; Mentor Graphics:n PADS sekä Altium Designer eli aiemmin Protel:na tunnettu. Kummalla tahansa päästäisiin samaan lopputulokseen, mutta suunnittelijan käyttökokemusten ja mieltymysten perusteella päädyttiin käyttämään ensin mainittua. PADS ohjelmisto tarjoaa omat moduulinsa niin piirikaavion suunnitteluun (PADS Logic), komponenttien sijoitteluun (PADS Layout) sekä varsinaisen piirilevyjohdotuksen toteuttamiseen (PADS Router). Kaikkien ohjelmiston moduulien toiminta linkittyy toisiinsa varsin johdonmukaisesti, joka helpottaa ja nopeuttaa suunnittelutyötä. Myös PADS tarjoaa työkalut Signal Integrity testeihin, mutta aiemmin mainitusta syystä tälle ei ole tarvetta.

4.2. Piirikaaviot ja komponenttivalinnat

Toteutettava keskusyksikkö on fyysiseltä rakenteeltaan hieman tavallisesta poikkeava. Ensimmäinen suunniteltava versio sisältää sekä ”kovalle” piirilevyille (FR-4) että joustavalle piirilevyille (esim. polyimidi) toteutettuja osioita. Elektroniikassa tällaista rakennetta kutsutaan yleisesti flex –rigid rakenteeksi. Rakenteesta johtuen keskusyksikkö on syytä jakaa toiminnallisiin lohkoihin jo piirikaavion suunnitteluvaiheessa suunnittelun helpottamiseksi. PADS Locigilla tämä onnistuu jakamalla kaavio välilehdille, jolloin kukin osakokonaisuus saadaan selkeästi muokattavaksi samanaikaisesti ja samassa näkymässä. Sopivia osakokonaisuuksia

toiminnallisuuden perusteella ovat varsinainen mikrokontrolleriosio oheiskomponentteineen, Bluetooth -moduuli oheiskomponentteineen ja järjestelmän tehonsyötöstä huolehtiva power -osio akkuineen. Lisäksi suunnittelussa tulee ottaa huomioon varaus ANT -radiomoduulille, jonka toimittaa eräs projektin osapuolista toimintavalmiina.

Keskusyksikön varsinainen prosessorimoduuli koostuu kahdesta erillisestä mikrokontrollerista sekä näiden oheiskomponenteista ja on näin ollen suurin keskusyksikön moduuleista. Molemmat kontrollerit ovat Atmelin AVR Atmega sarjasta ja ne on valittu erityisesti suorituskykyä ja tehonkulutusta silmällä pitäen. Tärkeä tekijä prosessoreita valittaessa oli myös kehitysalustojen toimivuus sekä saatavuus ja vaadittava alusta oli jo hankittuna aiempien projektien puitteissa. Ensimmäinen prosessoreista, Atmega644P, toimii järjestelmän ulkoisten ja sisäisten yhteyksien ylläpitäjänä. Kyseinen kontrolleri kykenee toimimaan peräti 14 MHz:n kellotaajuudella kun käyttöjännite on 3,3V. Tässä sovelluksessa kellotaajuudeksi valittiin kuitenkin 11,0592 MHz, jolloin käyttöjännitteen suuruudelle jää riittävä marginaali. Kontrollerin sisäinen väylänleveys ja tehonkulutus huomioon ottaen varsin tehokkaalla suorittimella on mahdollista suorittaa pienimuotoista datan prosessointia, kuten alkuperäisessä konseptisuunnitelmassa oli määritelty. Kontrollerilla on niin ikään toteutettu laitteistotasolla muun muassa kaksi USART -liityntää, jolloin sarjamuotoiset liittynät ulkoisille oheislaitteille voidaan toteuttaa luotettavasti. [17]

Toinen mikrokontrolleri, Atmega128L, toimii siltana edellä esitellyn kontrollerin (atmega644P) ja housuihin sijoiteltujen anturisolmujen välillä. Toteutustavaltaan kyseinen kontrollerimalli on hieman vanhempi ja jotta sitä voidaan käyttää 3,3V käyttöjännitteellä tulee valita juuri suorituskyvyltään hieman karsittu L -versio, kuten tässä on tehty. Tämän sillan suorituskykyvaatimus ei ole yhtä suuri kuin edellä esitellyn pääprosessorin, joten suoritin onkin asetettu toimimaan maltillisella 3,6864MHz:n taajuudella. Suurin tällä saavutettava hyöty onkin kellotaajuuden myötä laskenut virrankulutus. Tämän siltakontrollerin tarkoitus on muun muassa alustaa siihen kytketyn anturiverkon solmut isäntäprosessorilta saamiensa käskyjen mukaisesti, huolehtia anturiverkon ylläpidosta ja lähetysvuoroista (kyseessä on half-duplex verkko) sekä luovuttaa dataa eteenpäin pyydettyäessä.

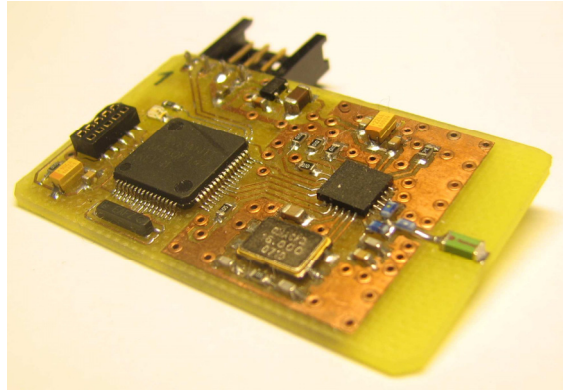
Prototyypivaiheessa erillisten mikrokontrollerien käyttö laskentaan ja anturisillan hallintaan oli välttämätöntä, sillä varsinaisen anturiverkon kehitys oli projektin aikana eriytetty laskentayksikön kehityksestä. Erillisten komponenttien käyttö mahdollisti näin ollen rinnakkaisen kehityksen, joka oli välttämätöntä kiireisen aikataulun puitteissa. Mahdollisen jatkokehityksen kannalta kahden mikrokontrollerin ratkaisun vaihtaminen yhteen kontrolleriin jätti selvän toiminnan ja toteutuksen optimointimahdollisuuden.

Molemmille kontrollereille kellosignaali tuotetaan erillisellä, perustaajuudella värähtelevällä rinnakkaisresonanssikiteellä (*engl. crystal*). Ulkoisten kiteiden käyttö voidaan perustella niillä saavutettavalla paremmalla tarkkuudella verrattuna piirin sisäiseen kelloon. Erityisesti laiteasolla toteutetut sarjamoitotiset liittymät ovat herkkiä taajuusvaihteluille, joita syntyy käytettäessä kontrollerin käyttöjännitteestä riippuvaa sisäistä RC -oskillaattoria. Oikean toiminnan takaamiseksi nämä kiteet vaativat rinnalleen ainoastaan niin kutsutut kuormituskondensaattorit, joiden suuruudeksi on valittu datalehden tietoihin perustuen 22pF.

Käytännössä keskussyksikkö sisältää kaksi erillistä anturisolua; toinen langallisille antureille ja toinen ANT -radiolla toimiville langattomille anturisoluille. Suuri osa järjestelmän osakokonaisuuksista toteutetaan hajautetusti eri projektiosapuolien kesken, joten oli järkevää käyttää yleisesti tunnettua datansiirtotekniikkaa. Kaikki kontrollereiden välinen dataliikenne keskussyksikössä tapahtuu näin ollen asynkronisella UART -sarjaliikenteellä. Kahden rinnakkaisen sillan fyysiset eroavaisuudet on pyritty kumoamaan järjestelmää suunniteltaessa. Toisin sanoen isäntäprosessorin pyytäessä anturidataa tietyllä solmutunnisteella anturiverkosta sen ei tarvitse tietää tuliko vastaus langalliselta vai langattomalta solmulta.

ANT -radio on, Bluetooth -teknologian tavoin, 2,4GHz:n ISM -kaistaa hyödyntävä lyhyen kantaman radioprotokolla, jolla voidaan muodostaa niin kutsuttu PAN (Personal Area Network). ANT -radio käyttää Nordic Semiconductorin nRF24AP1 RF -piiriä, johon Dynastream Innovations:n kehittämä ANT -protokolla on valmiiksi integroitu. Kyseinen piiri on suunniteltu pienivirtaiseksi ja kykenee parhaimmillaan toimimaan jopa vuosia yhdellä nappiparistolla. Protokolla käyttää aikajakoista kanavointia ja samanaikaisesti voi olla käytössä maksimissaan 4 samanaikaista kanavaa [18]. Esite käytetystä ANT -radiopiiristä on nähtävissä liitteessä 1.

HipGuard -järjestelmässä ANT -radioverkkoa käytetään pääasiassa datasiirtoon asusteen ja kenkien langattomille antureiden ja keskussyksikön välillä. Yksi ANT -radion kanavista on varattu liikennöintiin Suunnon valmistamalle T6 rannetietokoneelle. Valmiiksi ANT -protokollaa hyödyntävä T6 mahdollistaa jalan asentotietojen tai esimerkiksi varoitusten näyttämisen laitteen näytöllä. Kuvassa 4.1 on prototyypiversio ANT -radiomoduulin fyysisestä toteutuksesta HipGuard -järjestelmässä. Lopullinen versio tullaan liittämään joustavalla piirilevyllä varsinaiseen prosessorilevyyn. Tällöin liittymöiksi levyille riittävät TxD- ja RxD -linjat sekä käyttöjännitteet.



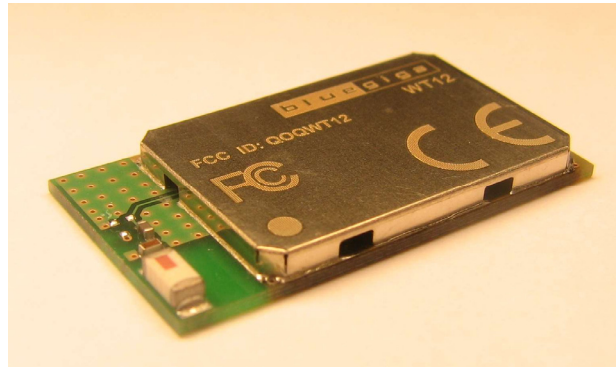
Kuva 4.1. ANT-radiomoduuli

Sekä vastaanotto- että lähetyspään ANT -radiomoduulit ovat fyysiseltä toteutukseltaan samanlaisia, jolloin solmujen master/slave -toimintamallit saadaan asetettua ohjelmallisesti. Isäntäprosessorina levyllä toimii Texas Instrumentsin 16 - bittinen MSP430F149, joka keskustelee ANT -protokollan tarjoavan nRF24AP1 radiopiirin kanssa niin ikään UART liitynnällä. Näiden integroitujen piirien lisäksi levyllä tarvitaan ainoastaan 3.3V regulaattori oheiskomponentteineen, 16MHz kide radiopiirille sekä passiivikomponenteilla aikaansaatava sovitus pala -antennille.

Liityntä langallisen anturisolmujen välillä tapahtuu sarjamuotoisella RS485 -liitynnällä. Erona tässä RS232 -liityntään on tapa, jolla loogiset bittitasot esitetään fyysisellä linjalla. RS485 -linjalla dataa siirretään differentiaalisesti kahta linjaa hyväksi käyttäen. Näin fyysiseen linjaan mahdollisesti vaikuttavat yhteismuotoiset häiriöt saadaan kumoutumaan vastaanottopäässä. Sovelluksen kannalta selkein saavutettava etu on kuitenkin RS485:n väylätyyppinen toimintaperiaate. Tämän ansiosta samaan fyysiseen linjaan voidaan liittää jopa 32 laitetta ja ohjelmistolla pystytään varmistamaan, ettei törmäyksiä siirtolinjalla synny. Selkeä negatiivinen puoli RS485 -tyypin verkossa on sen yksisuuntaisuus eli kyseessä on niin kutsuttu half-duplex verkko. Tästä toimintamallista johtuen RS485 linjan liitynnät on nimetty RS232 liitynnän RxD- ja TxD -linjoista poiketen A ja B linjaksi. Laitetasolla tämä fyysinen siirtolinja on toteutettu Analog Device:n komponentilla AD3483. Kyseinen komponentti ei välttämättä vaadi lainkaan ulkoisia oheiskomponentteja toimiakseen.

Keskusyksikköön toteutetaan langaton liityntä myös järjestelmästä ulospäin. Bluetooth -radiotekniikka valittiin käytettäväksi tähän lähinnä yleiskäyttöisyytensä ansiosta. Samalla Bluetooth -moduulilla voidaan muodostaa yhteys tavalliseen Bluetoothilla varustettuun PC:hen, matkapuhelimeen sekä lukuisiin muihin päätelaitteisiin. Koska Bluetooth -teknologia ei varsinaisesti ollut tämän järjestelmän toteutuksen tutkimusosa -alueena päädyttiin käyttämään valmista moduulia. Suomalaisen Bluegiga Technologiesin tarjoama vaihtoehto osoittautui kustannuksiltaan ja tekniseltä tueltaan lähes ylivoimaiseksi vaihtoehdoksi. Valmiin WT12 -moduulin

radio toimii luokassa 2 ja tukee 2.0+ EDR tiedonsiirtonopeuksia. Moduulissa on käytettävissä UART- ja USB -liittynät isäntäprosessoreiden liittämiseksi. Näistä UART-liityntä osoittautui järkeväksi vaihtoehdoksi omaan sovellukseemme ottaen huomioon piirillä ennestään käytettävissä olevat jännitetasot (3,3V). WT12 -moduulia ohjataan UART-liitynnän kautta erityisillä ASCII -merkkeihin pohjautuvilla iWrap -käskyillä, joihin perehdytään tarkemmin ohjelmisto –osiossa [19]. Alla olevassa kuvassa 4.2 on esillä WT12 Bluetooth -moduuli ennen varsinaiselle piirilevylle liittämistä. Esitekyseisestä moduulista on nähtävissä liitteessä 1.



Kuva 4.2. Bluegiga WT12 -moduuli

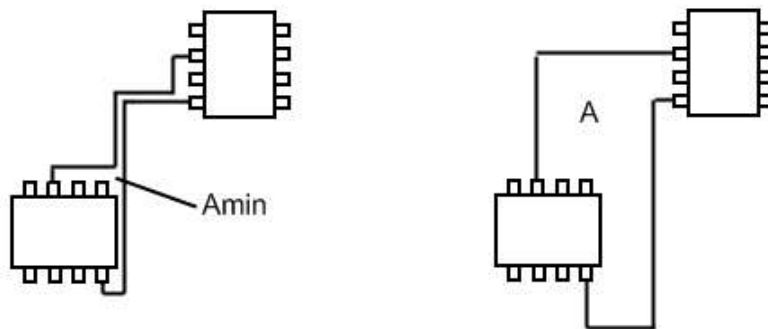
Teho järjestelmään saadaan Nokian BL-4C akusta, jonka kapasiteetti on suuruudeltaan luokkaa 800 mAh. Omaa latauspiiriä ei järjestelmään katsottu tarpeelliseksi integroida. Langattomat latausmenetelmät ovat vielä kehitysasteella ja mobiilikäyttöisen, päällepuettavan laitteen lataaminen johdollisella periaatteella olisi joka tapauksessa epäkäytännöllistä. Sen sijaan keskityttiin saamaan akunvaihtoprosessi mahdollisimman vaivattomaksi ja nopeaksi sekä toteuttamaan laite niin vähän virtaa kuluttavaksi kuin mahdollista vaihtotarpeen minimoimiseksi. Keskusyksikön kotelon taakse jysytyyn tilaan akun vaihto onnistuu ilman työkaluja yksikön ollessa irrotettuna asusteesta. Keskusyksikölle reguloidaan 3,3 V käyttöjännite erillisellä regulaattorilevyllä, jossa itse regulaattorina toimii National Semiconductorin LP2985-3,3. Regulaattorin luokiteltu virrankesto on 150 mA ja 50 mA:n kuormitusvirralla pudotusjännite (*engl. drop-out voltage*) ainoastaan noin 110 mV. Juuri alhainen pudotusjännite tekee komponentin hyvin soveltuvaksi akkukäyttöiseen järjestelmään, jossa akun nimellisjännite on 3,7V. Oikean toiminnan takaamiseksi regulaattori vaatii sisäänmenoonsa ja ulostuloonsa riittävän määrän kapasitanssia jännitteitä tasaamaan. Reguloidun jännitteen lisäksi akulta johdetaan suora syöttö langallisille anturisolmuille, jotka tästä edelleen reguloivat oman toimintansa edellyttämät jännitteet. Läheisesti tehonsyöttöön liittyvät myös integroitujen piirien käyttöjännitesuodatuksset. Käytännössä pienivirtaisille piireille sopivaksi havaittu suodinkondensaattorin suuruus on noin 100 nF. Joskus halutaan käyttää kahta erisuuruista kondensaattoria, jolloin kapasitanssiarvojen suuruudet valitaan noin dekadin jakovälillä.

Valmiit piirikaaviot komponenttivalintoineen ovat kokonaisuudessaan nähtävissä liitteessä 2.

4.3. Piirilevysuunnittelu ja periaatteet

Koska prototyypijärjestelmän piirilevyt tullaan toteuttamaan käsityönä, rajoittuu käytettävissä olevien johtavien piirilevykerroksien määrä kahteen. Tavoitteena suunnittelussa on hyödyntää molempia käytettävissä olevia tasoja niin, että muun muassa digitaalisen järjestelmän ylikuulumiset saadaan minimoitua, kuitenkin muuta toiminnallisuutta unohtamatta. Levyjen testattavuuteen ei tarvitse kiinnittää erityistä huomiota kunhan jokaiseen sähköiseen kontaktiin on mahdollisuus kytkeytyä esimerkiksi oskilloskoopilla.

Hyvän piirilevyn suunnittelu alkaa komponenttien sijoittelulla toiminnallisiin ryhmiin. Hyvä komponenttisijoittelu mahdollistaa reititysvaiheessa muun muassa piirilevyn virtasilmukoiden minimoinnin. Silmukoiden minimoinnista on etua ajateltaessa sekä piirilevysilmukan aiheuttamien häiriöiden määrää että piirilevyn häiriöherkkyyttä. Esimerkiksi integroitujen piirien vaatimat käyttöjännitesuodattimet kannattaa sijoittaa mahdollisimman lähelle itse piiriä, jolloin piirilevyjohdotuksista muodostuvat virtasilmukat minimoituvat lähes luonnostaan. Pinta -alaltaan suurempi virtasilmukka aiheuttaa paitsi ympäristöönsä haitallista differentiaalista häiriösäteilyä on myös itse herkempi ympäristöstä saapuville häiriöille. Alla olevassa kuvassa 4.3 vasemmalla on suositeltava tapa, jolloin silmukka -ala minimoituu. Oikealla on esimerkki huonosta johdotustavasta [20].



Kuva 4.3. Virtasilmukoiden minimointi.

Järjestelmän tuottamat kello-signaalit ovat luonteeltaan kanttiaaltoja ja merkittäviä häiriölähteitä jo suhteellisen pienillä kellotaajuuksilla. Syntyvien häiriöiden suuruus on riippuvainen paitsi kello-signaalin muutoksien esiintymistiheydestä (taajuus) myös signaalin nousevien ja laskevien reunojen muutosnopeudesta. Tämän päivän kehittyneimmät menetelmät tarjoavat mahdollisuuden nopeiden signaalien

muutosreunojen ”pyöristämiseen” (*engl. spread-spectrum*), mikä ei kuitenkaan ole mahdollista käyttämillämme komponenteilla. Syntyvien häiriöiden vaikutusta toisiin piireihin pyritään sen sijaan vähentämään, silmukka -alojen minimoinnin ohella, kasvattamalla kriittisten piirilevyvetojen etäisyyttä toisistaan. Tällöin ensimmäinen rajoittava tekijä on piirilevyn fyysinen koko eli etäisyyttä ei tavallisesti voida kasvattaa kovinkaan paljoa. Etäisyyden kasvattaminen vähentää merkittävästi myös ylikuulumista eli signaalin kytkeytymistä fyysisesti lähellä kulkevaan samansuuntaiseen johtimeen.

Digitaalisen järjestelmän maadoitusmenetelmä on aina maataso. Maataso tarjoaa aina signaalien paluuvirralle lyhimmän mahdollisen paluureitin lähtöpisteeseensä. Poikkeuksia tähän käytäntöön kannattaa tehdä ainoastaan jos halutaan ohjata suuria yksittäisiä paluuvirtoja erityisen herkkien komponenttien ohitse. Maapotentiaalin puhtauteen tietyillä alueilla kannattaa kiinnittää erityistä huomiota ja muun muassa käyttöjännite kannattaa reguloida mahdollisimman puhtaasta maasta. Jos järjestelmä sisältää digitaalisen maan lisäksi analogiamaan on ensiarvoisen tärkeää yhdistää nämä ainoastaan yhdestä pisteestä piirilevyllä. Maadoitusta suunniteltaessa maaajohdin kannattaa usein ajatella vastukseksi, mikä saattaa selkiyttää kokonaisuutta. [21]

Ensimmäisenä piirilevyille reititetään kriittiset datansiirtolinjat, joita keskussyksiköstä lähtee kohti kolmea kriittistä osakokonaisuutta; Bluetooth, ANT -verkko ja anturiverkko. Näin syntyvien läpivientien määrä tärkeimmissä piirilevyjohdotuksissa saadaan mahdollisimman pieneksi. Keskussyksikköön suunniteltiin myös varaus SD -muistikortin liittämiseksi, jolloin dataa siirretään huomattavasti nopeammalla (noin 2 MHz) nopeudella sarjamuotoisella SPI -tiedonsiirrolla. Kyseisen liittymän johdotus on suunniteltu mahdollisimman lyhyeksi. Vasta näiden jälkeen reititetään keskussyksikön käyttöjännitteeseen ja laitteen käyttöliittymään liittyvät johdotukset.

Valmista, reititettyä piirilevy-suunnitelmaa voidaan verifioida PADS ohjelmiston sisältämällä DRC (Design Rule Check) työkalulla. Työkalun osioista voidaan hyödyntää ainakin Clearance check:ä, joka tarkastelee että suunnittelijan ennalta määrittelemät eristevälit toteutuvat eri puolilla piirilevyä. Suurin vastuu levyn onnistumisesta on kuitenkin suunnittelijalla, joten automaattisia menetelmiä kannattaa käyttää korkeintaan suunnittelun tueksi.

4.3.1. Erityishuomioita taipuisille piirilevyille

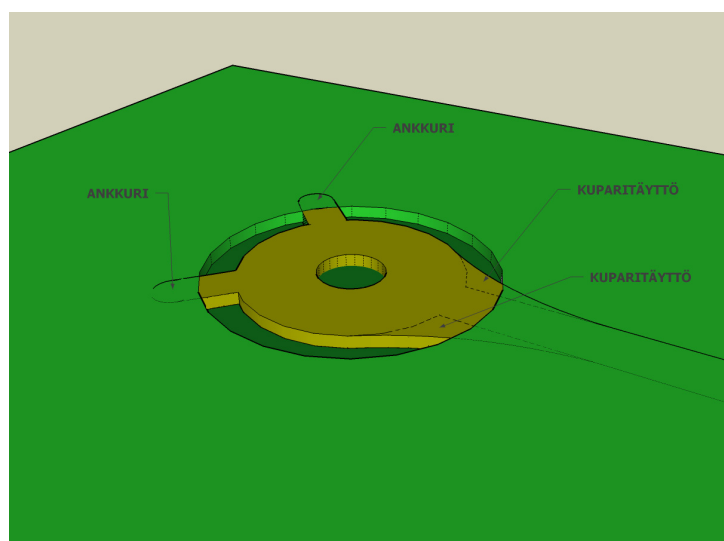
Taipuisat piirilevyt eivät ole uusi keksintö, sillä niitä tiedetään olleen varsin laajamittaisesti käytössä jo toisen maailmansodan aikaan 1940 -luvulla. Tämän jälkeen muutamia vuosikymmeniä käyttöaste pysyi lähes samalla tasolla ollen kuitenkin merkittävässä osassa muun muassa avaruusteknologioita kehitettäessä. Lopulta 70 -luvun taitteessa etenkin japanilaisyritysten toimesta taipuisien levyjen käyttö alkoi yleistyä ja onkin viime vuosina ollut yksi voimakkaimmin markkina -asemiaan

vahvistava teknologia elektroniikassa. Lisäksi kasvun on ennustettu jatkuvan lähivuosina lähes samalla tasolla, lähinnä johtuen valmistajien ja kuluttajien erityisestä kiinnostuksesta taipuisaa teknologiaa kohtaan. [22]

Usein taipuisia piirilevyjä kuulee kutsuttavan myös joustaviksi piirilevyiksi, vaikka eivät sellaisia olekaan sanan varsinaisessa merkityksessä. Taipuisien piirilevyjen suunnittelu on vielä tälläkin hetkellä varsin voimakkaasti kehittyvä ala, jota ylläpitävät kehittyvät materiaalit ja tuotantomenetelmät. Tietyiltä osin taipuisien piirilevyjen suunnittelumetodit poikkeavat merkittävästi perinteisten ”kovien” levyjen suunnittelusta, joten tutustumaan näihin hieman tarkemmin. Oikealla suunnittelulla voidaankin tehtyjen tutkimusten mukaan parantaa merkittävästi taipuisien piirilevyjen taivutuskestävyyttä. [23]

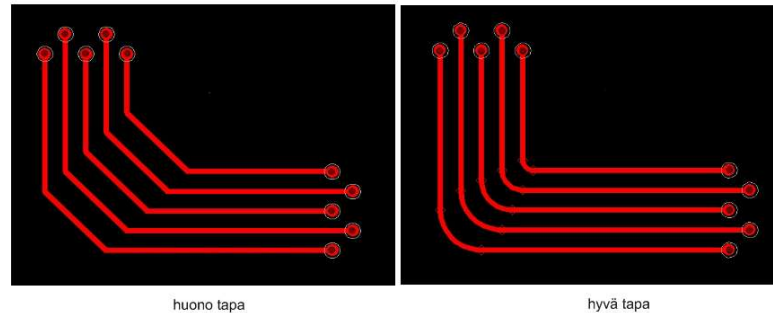
Taipuisia piirilevyjä valmistettaessa kupari kiinnittyy substraattimateriaaliin kuten tavallisilla FR-4 piirilevyilläkin. Komponenttien liittäminen juottamalla ja piirilevyn taipumisesta aiheutuvat rasitukset heikentävät kuitenkin kiinnittymistä ajan myötä. Luotettavuutta voidaan parantaa merkittävästi muutamilla pienillä pädeihin kohdistuvilla lisäyksillä, joten tutustutaan seuraavaksi muutamiin tärkeimmistä.

Ankkurit (*engl. tie-downs*) ovat pieniä kuparialueita, joiden tarkoitus on ulottaa padin kuparialue muutamasta kohdasta suojapinnoitteen (lyh. JEP) alle ulottuvaksi ja näin parantaa kiinnitystä. Kuparitäyttö (*engl. filleting*) padin ja vedon yhtymäkohtaan vähentää pisteittäistä rasitusta ehkäisten näin murtumia ja kuparin irtoamista substraatista. Seuraavasta kuvasta 4.4 on hyvin nähtävissä ankkureiden sijoittuminen suojapinnoitteen alle (vihreä alue) sekä kuparitäyttö padin ja vedon yhtymäkohtaan. [24]



Kuva 4.4. Esimerkki hyvästä pad:stä.

Käyttötarkoituksestaan johtuen taipuisa levy voidaan karkeasti jaotella kahteen osaan; taivutus- ja ladonta -alueisiin. Taivutusalueita suunniteltaessa tulee ensisijaisesti välttää johdinleveyden muuttamista ja johdotuksen suunnan muutoksia. Jälkimmäinen osoittautuu usein kuitenkin hankalaksi, jolloin suositellaan käytettäväksi tasaisesti kaartuvia johdotuksia terävien kulmien sijasta. Alla oleva kuva 4.5 esittää visuaalisessa muodossa huonon ja suositellun tavan piirilevyvetojen toteuttamiseen taivutusalueella.



Kuva 4.5. Piirilevyjohdotukset taivutusalueella.

Taipuisien yksikerrospiirilevyjen taivutussäteen ohjeellisena raja -arvona voidaan pitää 3 - 6 kertaista piirilevyn materiaalivehvuutta. Toisin sanoen tätä lukua pienempää taivutussädettä piirilevymateriaali ei tavallisesti kestä ja tiukemman taitoksen syntyminen on syytä estää esimerkiksi mekaanisesti rajoittamalla. Taipuisia piirilevyjä on tarjolla useita vahvuuksia, kuitenkin yleisimpänä 35 μm kuparikerros 50 μm eristemateriaalilla. Tällaisen piirilevymateriaalin pienin taivutussäde on näin ollen suuruusluokkaa 510 μm , joskin taivutussäde kannattaa aina suunnitella mahdollisimman suureksi. Taipuisien piirilevyjen tutkimus on edelleen aktiivisena tutkimusalueena ja taivutuskestävyydestä onkin viime vuosina julkaistu lukuisia tutkimuksia. [23] [24]

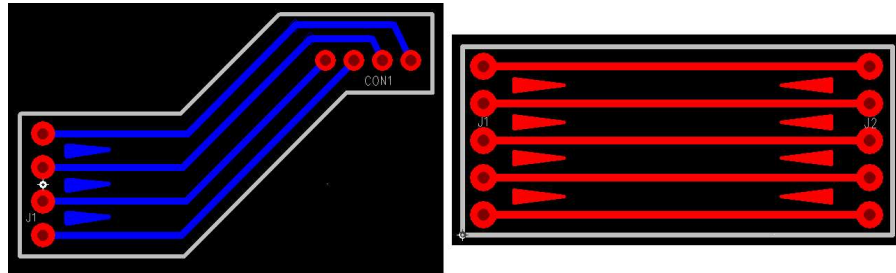
Taipuisan piirilevyn ladonta -alueille voidaan fyysisesti erityisen herkkien komponenttien, kuten pienijalkaisten IC:den, alle kiinnittää erilaisia kovikkeita (*engl. stiffeners*) kiinnitystä tukemaan. Jäykisteitä käytettäessä tulee kiinnittää erityistä huomiota piirilevyn pinnalla olevan suojapinnoitteen ja piirilevyn alle sijoitetun jäykisteen riittävään limitykseen. Hyvänä sääntönä voidaan pitää vähintään 1.0 mm:n limitystä, joka jo estää mahdollisten rasitus- ja murtumapisteiden syntymistä. [24]

4.4. Valmiit piirilevy suunnitelmat

Keskusyksikön valmiiden piirilevy suunnitelmien osalta tutustutaan tarkemmin toteutettuihin taipuisiin piirilevyihin, joiden tarkoitus on siis toimia liityntänä kovien FR-4 levyjen välillä. Varsinaisten kovien piirilevyjen toteutuneet layout -suunnitelmat

ovat varsin perustavaa laatua ja ne ovat nähtävillä pääpiirteissään jäljempänä kappaleessa 6.3 käsiteltäessä laitteen kokoonpanoa.

Prototyypivaiheessa flex – rigid -rakenne tullaan toteuttamaan käsityönä, joten tätä varten kehitettiin oma liitostekniikka taipuisien ja kovien piirilevyjen liittämiseksi toisiinsa. Piirilevyjohdotukset (*engl. routing*), joilla taipuisat piirilevyt lopulta toteutettiin, muodostuivat kovien piirilevyjen liityntöjen vaatimuksien perusteella seuraavanlaisiksi:



Kuva 4.6. Taipuisat piirilevyt vahvistuksineen.

Yläpuolella kuvassa 4.6 esitellyissä piirilevyissä pyöreinä punaisina alueina näkyvät piirilevyjen pad:it tullaan liittämään kovilla piirilevyillä sijaitseviin vastinpisteisiin. Suunnitelmista on hyvin nähtävissä menetelmä, jolla piirilevyvetojen kestävyyttä pyritään parantamaan erityisesti kohdassa, jossa jäykkä FR-4 piirilevy ei enää tue taipuisaa levyä. Kolmion muotoon piirrettyjen kuparialueiden tarkoituksena on asteittain helpottaa piirilevyn taipumista siirryttäessä jäykän piirilevyn reuna -alueelta pois päin. Näin mahdollisen taivutuksen yhteydessä taivutus saadaan tapahtumaan pääosin levyn keskellä, eikä niinkään murtumille herkällä kovan ja taipuisan piirilevyn yhtymäkohdassa. Kyseisen menetelmän mahdollisesta hyödystä ei vielä suunnitteluhetkellä ollut täyttä varmuutta, sillä se oli kokeiluluentoisena käytössä ensimmäistä kertaa.

Ajatuksena kovien ja taipuisien piirilevyjen välisen sähköisen kontaktin luomisessa sekä fyysisessä kiinnityksessä on hyödyntää kaupallisesti myynnissä olevaa piirilevyläpivientien holkitukseen tarkoitettua työkalua. Tähän niin ikään ensimmäistä kertaa koekäytössä olleeseen menetelmään paneudutaan hieman tarkemmin seuraavassa kappaleessa, joka käsittelee keskusyksikön toteutusta fyysisellä tasolla.

5. KESKUSYKSIKÖN TOTEUTUS JA INTEGROINTI

5.1. Piirilevy liitosalustana

Liitosalustan, tässä tapauksessa piirilevyn, perustehtävänä on toimia tukialustana komponenteille sekä toteuttaa vaadittavat siirtotiet signaaleille ja tehonsyötölle. Varsinkin radiopiirien ja muiden suuritaajuuksisten järjestelmien yhteydessä liitosalustalla on lisäksi usein tehtävänä osallistua komponentin suojaukseen esimerkiksi sähkö- ja magneettikentiltä ja toisaalta suojata ympäristöä komponenttien aiheuttamilta säteilykentiltä. Niin ikään komponentin suojaaminen mekaanisilta rasituksilta on usein liitosalustan vastuulla. Usein kova liitosalusta tarjoaa myös mahdollisuuden integroida osa passiivikomponenteista (vastukset, kondensaattorit, kelat) suoraan liitosalustaan, jolloin piirin koko pienenee ja luotettavuutta heikentävien juotosten määrä minimoituu. Taipuisat piirilevyt sen sijaan eivät sinällään tarjoa hyvää mekaanista suojausta, mutta ongelma on kierrettävissä erityyppisillä jäykisteillä. [25]

5.1.1. Piirilevymateriaalit

Tällä hetkellä selkeästi käytetyimpiä alustamateriaaleja elektroniikan valmistuksessa ovat orgaaniset liitosalustat. Näitä alustoja valmistetaan sekä taipuisista että jäykistä materiaaleista. Yleisin jäykkä alustamateriaali on epoksilasikuitulaminaatti eli FR-4. Muiden käytettävissä olevien kovien liitosalustojen ongelmaksi muodostuu useimmiten kustannukset vaikka ominaisuudet muilta osin olisivat erinomaiset. Esimerkki hyvästä materiaalista on syanaattiesterilaminaatti, jonka ominaisuudet muun muassa eristevakion ja häviökertoimen suhteen ovat FR-4:ää paremmat, mutta kustannukset rajoittavat käyttöä. [22]

Tavallisimmin käytetyt taipuisat piirilevymateriaalit ovat polyimidi ja polyesteri. Polyesteri on piirilevymateriaalina edullinen, mutta kestää huonosti juotosprosessia. Edullista polyesteria käytetään kuitenkin jossain määrin toteuttamalla piirilevyjohdotus polymeeripaksukalvojohtimilla ja käyttämällä komponenttien liittämässä johtavia liimoja. Uusi tulokas laminaattimateriaalien joukossa on PEN (*engl. Polyethylene naphthalate*), joka sijoittuu ominaisuuksiensa ja kustannustensa puolesta juuri aiemmin mainittujen välimaastoon. Niin ikään mielenkiintoinen vaihtoehto varsinkin puettavien järjestelmien kannalta on LCP -laminaatti (*engl. Liquid Crystal Polymer*), jonka

kosteudensieto -ominaisuudet ovat omaa luokkaansa. Kyseistä materiaalia käytettäessä polyimidilaminaatille välttämätön kosteutta poistava esilämmitysvaihe voidaan jättää pois. Lisäksi LCP soveltuu keskimääräistä paremmin suuritaajuisten järjestelmien alustaksi dielektrisyysvakionsa ansiosta. Alla oleva taulukko 5.1 kokoaa yhteen yleisimpien taipuisien piirilevy materiaalien ominaisuudet. [22] [24]

Taulukko 5.1. Taipuisat piirilevy materiaalit

Substraatti	Dielektrisyysvakio	Eristyskyky	Kosteuden absorbointi	Vetolujuus
Polyesteri	3,2	7000V/mil	< 0,08%	25 kpsi
Polyimidi	3,5	7000V/mil	1,3 - 3,0%	25 kpsi
PEN	2,9	7500V/mil	1 %	30 - 35 kpsi
LCP	2,9	6000V/mil	0,02 - 0,1%	15 -25 kpsi
PVC	4,7	500V/mil	< 0,5%	5 kpsi

5.2. Piirilevyvalmistus

Teollinen fotoresistikalvoon perustuva piirilevyvalmistus poikkeaa merkittävästi käsityönä prototyypivaiheesta tapahtuvasta valmistuksesta. Teolliseen valmistukseen perehtyminen ei kuitenkaan ole olennaista järjestelmän toteutuksen kannalta varsinkaan prototyypivaiheessa. Jos piirilevyllä käytettävät eristevälit ovat riittävän suuria, voidaan usein varsinkin prototyypivaiheessa piirilevy toteuttaa esim. mekaanisesti jyrsimällä. Tähän soveltuvan laitteen hankintakustannukset usein huomattavan kalliita tarkkuusvaatimuksista johtuen ja tällöin ainoaksi vaihtoehdoksi jää valotus/syövytysmenetelmä.

Ilman suojaavaa lakkakerrosta mikä tahansa valmistettu piirilevy on erityisen arka ulkoisille haittavaikutuksille kuten naarmuuntumiselle ja varsinkin hapettumiselle. Kuparin hapettuminen heikentää liitoksen luotettavuutta varsinkin jos piirilevyvalmistuksen ja komponenttien juottamisen välinen aika muodostuu suureksi. Piirilevyjohdotusten osalta hapettumisongelma on ratkaistavissa juotteenestopinnoitteella, jolloin ongelmakohdiksi jäävät varsinaiset juotospinnat. Varastointikestävyyden parantamiseksi juotospinnat käsitelläänkin usein ennen varastointia kemiallisella tinalla. Nykyään kemiallinen käsittely on muuttunut vanhasta metaanisulfonipohjaisesta menetelmästä rikkihappopohjaiseen käsittelyyn, joka on prosessiltaan varmempi ja lisää mahdollisuuksia valmistukseen, toimien esimerkiksi johdinkuvion syövytysresistinä. Teollisessa piirilevyvalmistuksessa juotosten

luotettavuutta pyritään lisäksi parantamaan käyttämällä juotosuunissa erityistä typpiatmosfääriä.

HipGuard -järjestelmän keskussyksikön piirilevyt tullaan prototyypivaiheessa toteuttamaan syövyttämällä käsityönä. Tämä menetelmä rajoittaa luonnollisesti käytettävissä olevien reitityskerroksien määrän kahteen, sillä monikerroslevyjä on mahdollista tuottaa vain teollisesti laminointiperiaatteella. Kovat piirilevyt valmistetaan käyttäen 1,6mm vahvuista FR-4 piirilevyä, jonka kuparikerroksen vahvuus on 35 µm. Käyttäen mustesuihkutulostimella tulostettuja valotusmaskeja ja erityistä valotuslaitetta valotetaan piirilevyn pinnalla olevaa valoherkkää fotoresistikalvoa (*engl. photoresist*) noin 30 sekuntia.

Ylimääräisen, valottuneen fotoresistin poistaminen sekä kovilta että taipuisilta levymateriaaleilta tapahtuu käyttäen noin 0,7 % natriumhydroksidi -liuosta. Käytännössä 1 - 2 minuutin liottaminen kehitteessä on yleensä riittävä aika irrottamaan lakkapohjainen, valottunut fotoresisti. Riittävä kehitysaika on usein määritettävissä visuaalisesti eli toivottavaan tulokseen on päästy kun piirilevyjohdotus on selvästi näkyvillä aihiossa. Käytännössä on havaittu, että useimmiten taipuisa piirilevy huomattavasti tarkempi käytettävästä valotusajasta ja liiallinen valotus on jopa haitallinen lopputuloksen kannalta. Kehitteestä poiston jälkeen levyä on hyvä huuhdella runsaasti puhtaalla vedellä edessä olevaa syövytysvaihetta silmällä pitäen.

Kuparin syövytykseen on tarjolla muutamia vaihtoehtoisia kemikaaliyhdistelmiä. Tällä kertaa käytettiin vedestä (H₂O), suolahappoliuoksesta (HCl) sekä vetyperoksidista (H₂O₂) muodostettavaa liuosta. Sekoittaessa liuosta lisätään ensin neljään tilavuusyksikköön vettä kaksi tilavuusyksikköä noin 30 % suolahappoa, jonka jälkeen syövyte viimeistellään kolmella tilavuusyksiköllä niin ikään noin 30 % vetyperoksidiliuosta. Tyypiltään näin syntyvä syövyte on niin kutsuttua nopeaa syövytettä, jonka nopeutta voidaan kuitenkin hienosäätää käyttämällä erilämpöisiä lähdekemikaaleja. Nyrkkisääntönä toimii että lämmin syövyte toimii kylmää nopeammin. Varsinkin taipuisien levyjen yhteydessä liian nopea syövyte saattaa kuitenkin syövyttää kaiken kuparin, fotoresististä huolimatta. Kehitetty piirilevyaihio upotetaan syövytteeseen ja annetaan prosessin käynnistyä. Jälleen parhaana mittarina prosessin valmistumisen suhteen toimii visuaalinen tarkistus. Levy on valmis kun kaikki ei -toivottu kupari on syöplynyt levyltä. Syövytysreaktion ollessa käynnissä tarkistusta helpottaa levyn nostaminen syövytteestä, josta ei mainittavasti ole haittaa prosessin etenemiselle. Itse syövytysprosessin kesto saattaa vaihdella hyvinkin paljon käytetyistä materiaaleista riippuen. Nopeimmillaan levy saattaa syöpyä jopa 10 sekunnissa, toisaalta joskus prosessi saattaa kestää muutamankin minuutin. Syövytysparametrit kannattaakin usein hioa kohdalleen muutaman testikappaleen avulla jos prosessin toimivuudesta on epäilystä. Syövytetty levy on jälleen hyvä huuhdella runsaalla vedellä liuotejäämien poistamiseksi.

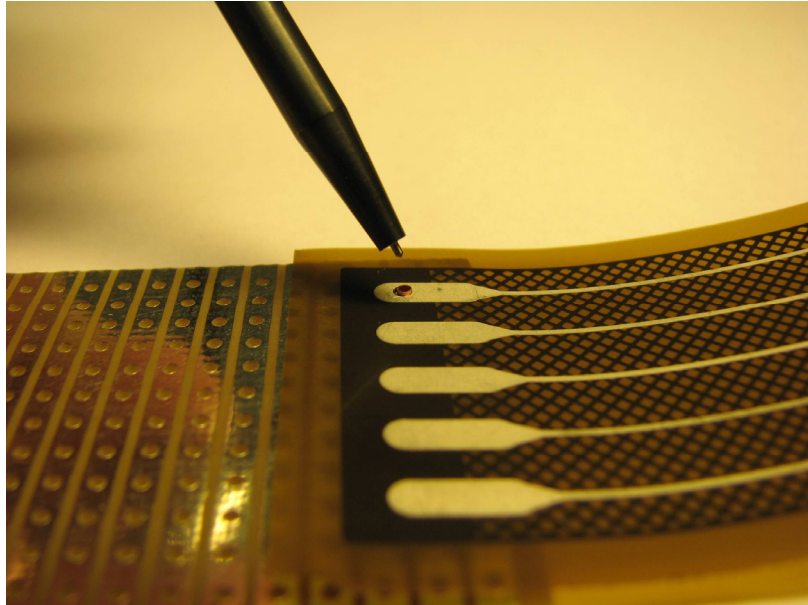
Levyn viimeistelyyn kuuluu jäljelle jääneen fotoresistin poistaminen, reikien poraaminen sekä mahdollinen kemiallinen tinapinnoite. Fotoresistijäämät poistetaan käyttäen jälleen valotuslaitetta, tällä kertaa ilman valotusmaskia ja lopuksi liottamalla kehitteessä. Kemiallinen tinapinnoitus kannattaa suorittaa vasta porauksen jälkeen, jolloin myös reikien kuparoinnit saadaan pinnoitettua. Läpivientireiät eivät kuitenkaan tällä keinoin tule läpipinnoitetuiksi (*engl. plated through hole*), vaan korvaava läpipinnoitus saadaan aikaan vasta läpivientien holkitustyökalulla.

5.3. Kokoonpano

Käsityönä toteutettava flex-rigid -rakenne on prototyypilevyn valmistuksen kannalta hankalin vaihe toteuttaa. Taipuisat piirilevyt liitetään koviin FR-4 piirilevyihin käyttäen piirilevyläpivientien holkitukseen tarkoitettua kaupallista työkalua, LPKFTM EasyContac:ia. Tällä kertaa sopivaksi holkkikooksi valittiin työkalusarjasta pienin, ulkohalkaisijaltaan 0.8 mm oleva holkkityyppi. Työkalun ohjekirjan perusteella tämä holkki vaatii 0.85 mm suuruisen reiän, mutta käytännössä asennusta helpottaa, jos käytetään 0.9 mm reikäkokoa. Sekä koviin että taipuisiin levyihin tulee siis porata samansuuruiset reiät.

Kokemuksen ja testien perusteella paras tapa toteuttaa reikien poraaminen on kohdistaa taipuisa levy kovan piirilevyn vastinpisteisiin ja varmistaa esimerkiksi teipillä asemoinnin säilyminen porauksen aikana. Näin samalla porauksella saadaan reikä aikaiseksi sekä taipuisaan että kovaan levyyn. Etenkin taipuisaa piirilevyä porattaessa on muutenkin syytä käyttää erityistä varovaisuutta, sillä ohut kuparipinta on herkkä irtoamaan substraattimateriaalista.

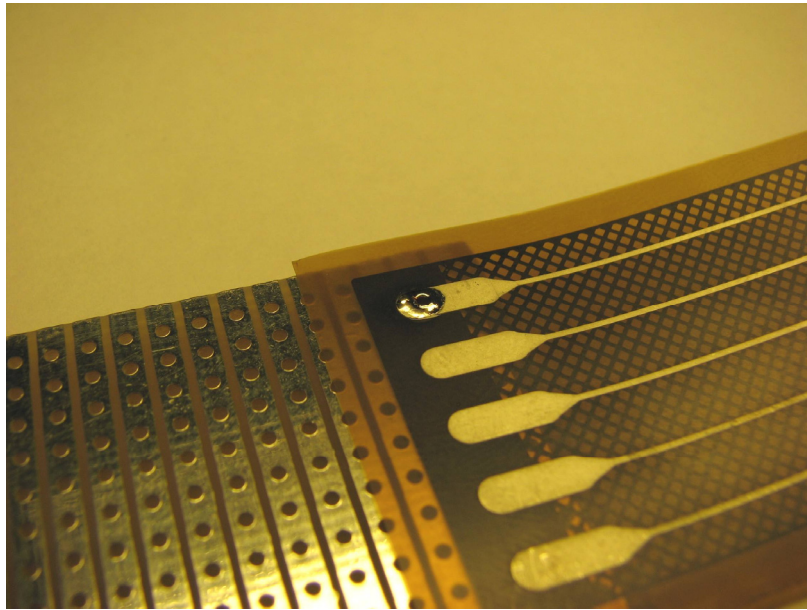
Holkkien kiinnittyminen piirilevyyn perustuu holkin toisen pään mekaaniseen levittämiseen erityistä työkalua käyttäen. Holkit asennetaan siten että sekä taipuisa että kova levy jää puristuksiin holkin väliin ja sähköinen kontakti muodostuu. Näin toimitaan jokaisen holkin kohdalla. Yhdellä toimenpiteellä saadaan siis aikaiseksi sekä fyysinen kiinnitys että sähköinen yhteys. Seuraava kuva 5.1 havainnollistaa toimenpiteen kulkua ja käytettävää työkalua.



Kuva 5.1. Flex-rigid -rakenteen holkitusvaihe.

Sähköisen kontaktin muodostuminen flex-rigid -liitoksessa pitää vielä varmistaa juottamalla. Myös tämä vaihe käsityönä edellyttää erityistä varovaisuutta ja käytännön ymmärrystä juottamiseen käytettävistä lämpötiloista. Taipuisan piirilevyn kuparialue lämpiää huomattavan nopeasti, koska ohut substraattimateriaali ei vähäisen massansa vuoksi kykene absorboimaan suuria määriä ”ylimääräistä” lämpöenergiaa. Pahimmillaan lämpötilan nousu liian korkeaksi aiheuttaa kuparin irtoamisen substraattimateriaalista ja levystä tulee käytännössä käyttökelvoton. Jälleen käytännön kokemusten perusteella sopivaksi juotoskolvin lämpötilaksi havaittiin noin 280 °C, lyijytöntä tinaa käytettäessä.

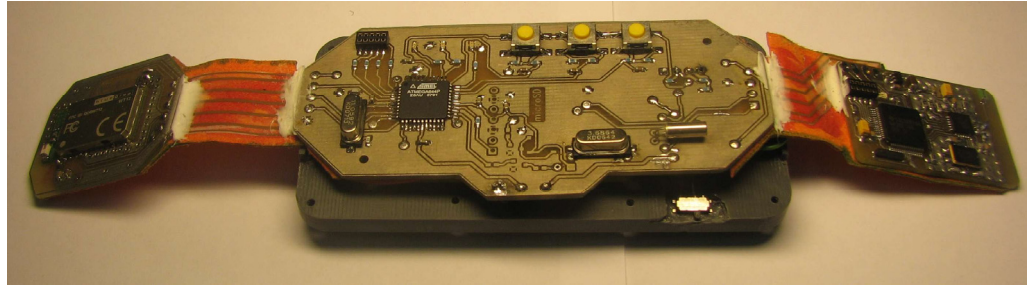
Seuraavassa kuvassa 5.2 on nähtävissä yhden kontaktin osalta valmis, juotoksella varmistettu flex-rigid -rakenne. Kyseisen tyyppisessä kiinnitysratkaisussa on käsityön osuus melko suuri ja menetelmä ei suoraan sovi siirrettäväksi edes pienimuotoiseen sarjatuotantoon riittävän nopeaksi. Näin ollen nopean ja luotettavan flex-rigid -rakenteen aikaansaamiseksi teollinen valmistus jää edelleen ainoaksi vaihtoehdoksi ja on varmastikin järkevää jos järjestelmä siirretään sarjatuotantoon.



Kuva 5.2. Valmis flex-rigid -rakenne

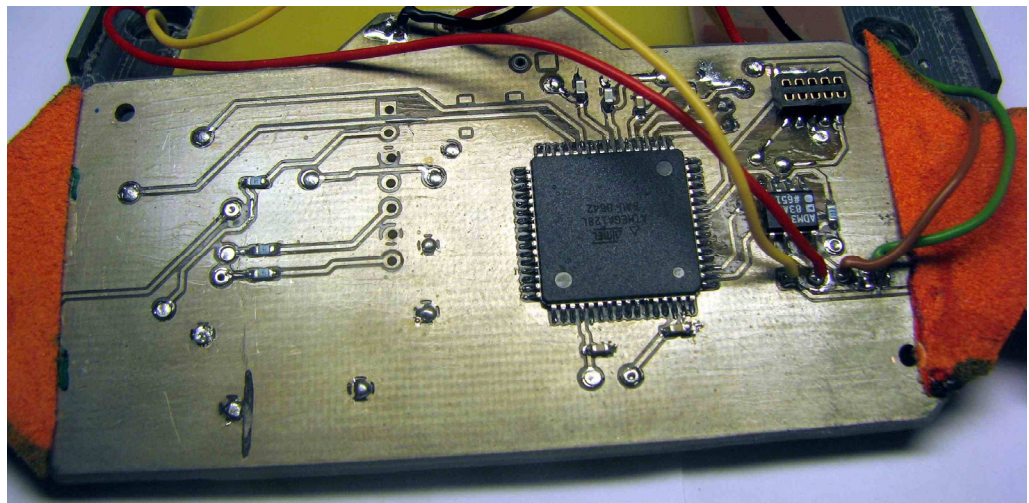
Muilta osin HipGuard -keskusyksikön elektroniikan kokoonpano oli varsin suoraviivainen operaatio. Komponentit juotettiin juoksutteella (*engl. flux*) käsitellyille piirilevyille käsityönä lyijytöntä tinaa käyttäen, tinakolvin lämpötilan ollessa 320 °C. Juoksutteen tehtävä on juotostapahtumassa puhdistaa lämmön avulla liitospintoja, jolloin juotoksesta saadaan kestävämpi. Vielä jonkin verran ongelmia tuotti omalle piirilevyllään rakennetun Bluetooth -moduulin juottaminen omalle piirilevyllemme. Kyseinen moduuli on suunniteltu ainoastaan reflow -uunissa tapahtuvaan juotosprosessiin ja tällöin moduulin pohjassa sijaitsevat sähköiset liittynät eivät ole lainkaan juottajan nähtävissä. Ongelma pyrittiin ratkaisemaan käyttäen käytössä olleen juotosaseman juotospuhallinta, jolla koko moduulin reuna -alue kuumennettiin mahdollisimman kuumaksi juottumisen aikaansaamiseksi. Keskusyksikön käyttötesteissä moduulin sähköiset kontaktit kuitenkin irtosivat useaan otteeseen, joten kiinnitysprosessissa oli selvästi parantamisen varaa. Mainittakoon, että moduulin valmistajalta ilmestyi projektin loppuvaiheessa vaihtoehtoinen tuote, jonka liittynät olivat selvästi paremmin käsillä käsityönä tapahtuvaa valmistusta ajatellen.

Keskusyksikön lopulliseksi pituudeksi radiomoduuleineen muodostui 215 mm ja leveydeksi leveimmästä kohdasta mitattuna 45 mm. Keskusyksikön valmis, juotettu piirilevy oheismoduuleineen on nähtävissä kuvassa 5.3. Taipuisien piirilevyjen alle on prototyypissä liimattu kangasjäykiste vähentämään testausvaiheessa piirilevyihin kohdistuvia kiertymisestä johtuvia rasituksia.



Kuva 5.3. Keskusyksikön valmis piirilevy etupuolelta.

Levyn keskellä on nähtävissä tilavaraus optionaalille microSD -muistikortille tilanteisiin, joissa halutaan suorittaa pitkäaikaismittauksia ilman kytkeytymistä anturointijärjestelmän ulkopuolelle älypuhelinta käyttäen. Kolmella kuvassa näkyvällä painonapilla saadaan keskusyksikköön aikaiseksi riittävä käyttöliittymä, jota tarkastellaan lähemmin kappaleessa 7.1.3. Alla on nähtävissä kuva 5.4 keskusyksikön piirilevystä takapuolelta katsottuna. Kuvassa näkyvä kontrolleri on anturisiltana toimiva AtMega128L.



Kuva 5.4. Keskusyksikön valmis piirilevy takaa.

6. KESKUSYKSIKÖN OHJELMISTO

6.1. Ohjelmiston suunnittelu

Tässäkin sulautetussa järjestelmässä laitteen ohjelmisto muodostaa merkittävän osan kokonaisuutta, joten sen suunnitteluun ja haasteiden tiedostamiseen kannattaa käyttää etukäteen riittävästi aikaa parhaan mahdollisen lopputuloksen ja ohjelmiston muokattavuuden tehostamiseksi. Käytännössä muokattavuus varmistetaan modulaarisella ohjelmistorakenteella eli jakamalla ohjelmiston toiminnallisuus omiin kokonaisuuksiinsa.

6.1.1. Ohjelmisto muuhun järjestelmään liitynnän kannalta

Käytännössä yhteydenpito keskusyksiköstä muuhun järjestelmään tapahtuu Bluetooth -tekniikan välityksellä. Järkevästi valittu Bluetooth -moduuli mahdollisti tällä kertaa mahdollisimman autonomisesti toimivan langattoman radiotekniikan. Bluetooth -moduulin omatoimisuudesta kertoo sekin, että käytännössä keskusyksikön ohjelmiston tulisi normaalisti toimiessaan asettaa moduulin asetusarvot ainoastaan kerran. Tämän jälkeen asetetut arvot säilyvät moduulin sisäisessä muistissa myös virtakatkoksista huolimatta. Varmuuden vuoksi ohjelmisto kuitenkin alustaa moduulin jokaisella käynnistyskerralla, näin vältetään ongelmatilanteilta siinäkin tapauksessa, että asetukset syystä tai toisesta resetoituisivat.

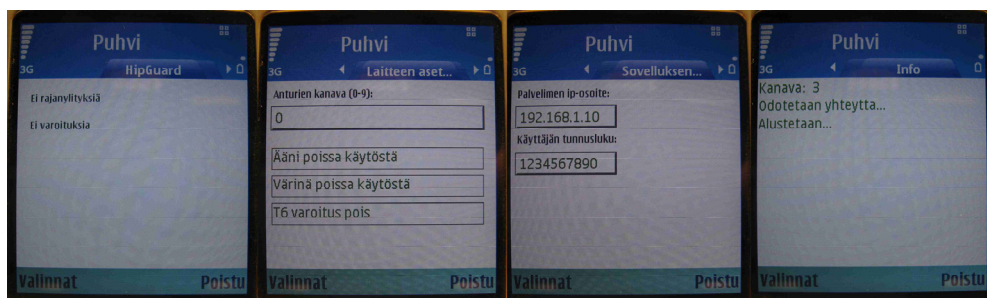
Ohjelmisto asettaa moduulin yrittämään yhteyttä mobiililaitteeseen noin 15 sekunnin välein. Yhteydenmuodostuksen onnistuessa dataliikenne toimii sarjaliikenneperiaatteella ilman, että moduulia tarvitsee erityisesti hallinnoida. Langattomat, lyhyenkin kantaman radiotekniikat muodostavat lähestulkoon poikkeuksetta potentiaalisen tietoturvariskin. Kevyen tason suojaa saatiin aikaiseksi konfiguroimalla WT12 -moduuli ja sitä myöten koko keskusyksikkö radiotasolla näkymättömäksi onnistuneen yhteydenmuodostuksen jälkeen.

Tällä kertaa järjestelmää ei ollut toteutettu hardware -tasolla soveltuvaksi vuonvalvontaan. Tämä huomattiin ongelmaksi myöhemmässä vaiheessa ohjelmistoa testattaessa erityisesti käytettäessä suuria datamääriä ja kaksisuuntaista liikennöintiä. Pitämällä tiedonsiirtonopeus kohtuullisena ongelmat voitiin kuitenkin kiertää.

6.1.2. Ohjelmisto käyttöliittymän kannalta

Järjestelmään toteutettiin kaksi erityyppistä ja -laajuista käyttöliittymää. Kattavamman käyttöliittymän järjestelmä tarjoaa sen ollessa yhteydessä älykkääseen mobiililaitteeseen, esimerkiksi matkapuhelimeen kuten edellisessä kappaleessa 7.1.1 lyhyesti esiteltiin. Keskusyksikön ja mobiililaitteen välisen yhteyden muodostuessa päivitetään ensimmäisenä puhelimelta keskusyksikölle senhetkiset aika- ja asetustiedot. Aikatiedon siirto tapahtuu yleisesti käytössä olevan UNIX -aikaleiman muodossa. Mobiililaitteen ja keskusyksikön välinen protokolla ja sen sisältämät parametrit on esitelty tarkemmin myöhemmin kappaleessa 7.2.2. Automaattisesti ajan tasalla pysyvän aikaleiman keskusyksikkö lisää jokaiseen kirjaamaansa tapahtumaan. Täten jokaiseen raja -arvon ylitykseen saadaan tieto todellisesta tapahtuma -ajankohdasta. Toisessa vaiheessa keskusyksikkö lähettää puhelimelle kirjaamansa tapahtumat ja puhelin esittää ne käyttäjälle ymmärrettävässä muodossa. Onnistuneen tietojen vaihdon päätteeksi puhelin lähettää yhteyden sulkemispyynnön keskusyksikölle ja jää odottamaan seuraavaa yhteydenottoa.

Puhelimeen oli siis tarpeen toteuttaa projektin puitteissa oma HipGuard -järjestelmää ymmärtävä käyttöliittymärajapinta. Puhelin käyttää Symbian Series 60 ohjelmistoalustaa, joten käyttöliittymä on toteutettu käyttäen Carbide.c++ -kehitysympäristöä. Esitellään käyttöliittymä lyhyesti käyttäen alla olevaa kuvaa 6.1 apuna käyttäen. Ohjelman päänäkymä kertoo käyttäjälle millainen tapahtuma ja mihin aikaan on viimeksi rekisteröitynyt järjestelmään. Välilehtiä hyödyntävän näkymän seuraava välilehti mahdollistaa järjestelmän asetuksien muuttamisen, kuten haptisen- ja äänipalautteen ottamisen käyttöön. Kuvan 6.1 kolmanteen näkymään voidaan asettaa mahdollisesti käytössä olevan verkkopalvelin IP -osoite ja kirjautumisen tunnusluku. Viimeinen Info -näkymä toimii konsolina, jonne ainakin kehitysvaiheessa tulevat esiin ohjelman suorittamat toiminnot.



Kuva 6.1. Nokia E60 HipGuard -käyttöliittymässä

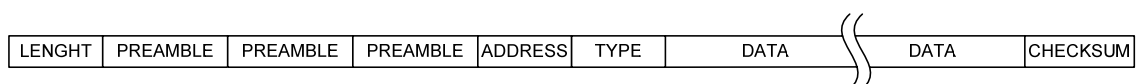
Kuten jo keskusyksikön määrittelyn yhteydessä esitettiin, päätettiin pääasiallista käyttäjäkuntaa silmällä pitäen keskusyksikön itsensä tarjoama käyttöliittymä pitää niin pelkistettynä kuin mahdollista. Tämä siksi, että lopputuotteen käyttäjistä suurin osa

tulisi varmasti olemaan vanhemman sukupolven edustajia. Käyttäjän antamat komennot saadaan keskussyksikköön kolmen painonapinvälityksellä. Olennaisempi osa tässä käyttöliittymässä on erivärisillä ledeillä toteutettu käyttäjän informointi. Vihreä ledi kertoo järjestelmän normaalista toimintatilasta ja että lonkan asento on sallittujen rajojen sisällä. Keltaisella ledillä järjestelmä ilmoittaa, että ollaan lähestymässä ennalta asetettuja raja -arvoja. Punaisella ledillä vastaavasti informoidaan käyttäjää vakavasta sisäisestä virhetilasta ja raja-arvon ylityksestä. Ledien lisäksi informointia voidaan tehostaa keskussyksikköön integroidulla piezo -summerilla, varsinkin tilanteissa jossa ledit ovat huonosti nähtävissä.

6.2. Kehysformaatti ja tiedonsiirtoprotokollat

Kaikki asynkroninen sarjamuotoinen UART liikennöinti järjestelmän sisällä tapahtuu samoilla verkon asetuksilla; nopeus 115200bps, 8 -databittiä, ei pariteettibittiä ja 1 -stop bitti. Näiden asetusten valintaa tuki erityisesti yleisesti käytössä olevien anturilaitteiden valmius toimia tarvittaessa kyseisillä asetuksilla. Vaikka datansiirron sinänsä tuleekin olla tässä sovelluksessa mahdollisimman luotettavaa, voitiin pariteettibitin tarkastelusta luopua sillä virheenvalvontaa suoritetaan aktiivisesti sovellustasolla.

Vaikka sarjamuotoiseen tiedonsiirtoon on kehitetty useita valmiita, periaatteessa kelvollisiakin ja standardoituja kehysformaatteja, päädyttiin silti omaan ratkaisuun. Omalla kehysratkaisulla saadaan paketeista karsittua pois kaikki tämän sovelluksen kannalta epäolennaiset elementit, jolloin käyttämämme suhteellisen pienikapasiteettisen verkon rasitus kevenee hieman. Vaatimuksena paketeille oli osoitteellisuus, paketin tyyppin määriteltävyys, vaihtuvamittainen hyötydatakenttä sekä tarkistussumma. Nämä vaatimukset saatiin täyttymään seuraavanlaisella kehysformaatilla.



Tarkastellaan hieman tarkemmin paketin kenttiä.

LENGHT -kenttä määrittelee koko datapaketin pituuden tavuina. LENGHT -tavua itseään ei lasketa mukaan tähän arvoon.

PREAMBLE -tavuja tulee esiintyä kelvollisessa datapaketissa kolme kappaletta peräkkäin. Yhden PREAMBLE -tavun sisältönä on aina heksadesimaaliarvoinen FF eli bitteinä pelkkää ykköstä. Vaikka sama kolmen tavun paketti saattaa esiintyä datan joukossa, kelpaa se silti hyvin synkronointi sekvenssiksi, jonka ympäriltä mahdollista kelvollista datapakettia etsitään. Tämä siksi, että kelvolliseen datapakettiin kuuluu muitakin ”vaadittavia” tavuja.

ADDRESS eli osoitekenttä. Master -laitteen lähettämissä paketeissa tämä 8 -bitin pituinen kenttä sisältää vastaanottajan (anturisolmu) osoitteen. Anturisolmujen

mahdolliset osoitteet ovat välillä "A"... "Z", joka siis tulee sisällyttää pakettiin heksadesimaalimuotoisena. Anturisolmujen generoimissa paketeissa tämä kenttä sisältää lähettäjän oman osoitteen, jolloin master saa tiedon paketin lähettäjästä. Myös broadcast -tyyppinen lähetys masterin osalta on mahdollinen ja tämä tapahtuu määrittelemällä osoitekentän arvoksi "b", niin ikään heksadesimaalimuodossa.

TYPE -kenttä määrittelee paketin tyyppin, jonka perusteella tiedetään mahdollisen datakentän sisällön merkitys. Arvot välillä 0x30...0x38 ovat mahdollisia. Tarkempi kuvaus tyypeistä on kappaleessa 7.2.1.

DATA -kenttä on paketin tyyppistä riippuen vaihtuvamittainen datakenttä. Tämä kenttä on optionaalinen, sillä kaikki pakettityypit eivät välttämättä sisällä varsinaista dataa (esimerkiksi positiiviset ja negatiiviset kuittaukset).

CHECKSUM -tarkistussummakenttä on myös pituudeltaan tavun mittainen. Kentän sisältö saadaan laskemalla yhteen paketin kaikki muut tavut paitsi LENGHT ja invertoimalla saatu summa. Tarkistussumma tulee myös tehdä 8 -bittiseksi "leikkaamalla" ylimääräiset eniten merkitsevät bitit pois ennen invertointia. Jos tarkistussumma ei täsmää, hylkää anturisolmu kyseisen paketin erikseen ilmoittamatta. [26]

6.2.1. Anturiverkon ja keskusyksikön välinen yhteyskäytäntö

Tiedonsiirto keskusyksikön laskentakontrollerin ja siltakontrollerin välillä on pakettipohjaista ja osoitteellista. Siltakontrollerin ohjelmisto on kuitenkin pyritty kehittämään siten, että sen olemassaolo ja toiminta on pääkontrollerille läpinäkyvää. Toisin sanoen järjestelmän käynnistyessä siltakontrolleri selvittää omatoimisesti peräänsä kytketyt anturisolmut ja näihin mahdollisesti kalustetut kiihtyvyyss-, magneetti- ja kapasitanssianturit.

Keskusyksikkö toimii anturiverkon master -solmuna ja anturisolmut slave -solmuina. Slave -solmut osoittavat aina lähettämänsä paketit masterille, joka muiden toimintojensa ohella hallitsee verkon ajoituksia. Tämän perusteella anturisolmut voivat siis lähettää dataa ainoastaan masterin sitä pyytäessä. Normaalisti käytetyn osoitteellisen liikennöinnin lisäksi myös masterilta tuotetut broadcast -lähetykset ovat mahdollisia. Jollei siis ole erityistä tarvetta viitata yksittäiseen anturisolmuun, kannattaakin paketit lähettää tässä muodossa. Tällainen paketti voisi olla esimerkiksi solmujen tiettyjen anturityyppien (magneettianturi) mittaustarkkuuden asetus. Jos broadcast -lähetys edellyttää vastausta anturisolmuilta, vastaavat kaikki laitteet vuorollaan. Tämä vuorollaan vastaaminen tapahtuu anturisolman avustuksella, sillä muussa tapauksessa päällekkäisten lähetysten todennäköisyys on suuri. Tämä siksi, että anturisolmut eivät ole tietoisia toistensa olemassaolosta ja osoitteista. Ohjelmistokehityksen kannalta pitää huomioida, että vaikka yksittäiseltä laitteelta vastaus saapuu tavallisesti alle kahdessa

millisekunneissa, voi vastausaika broadcast -pakettiin vaihdella hyvinkin paljon verkon koosta ja operaation tyypistä riippuen.

Pakettien liikennöintiä selkeyttämään ja tehostamaan määriteltiin käytettäväksi kahdeksan erilaista pakettityyppiä. Jokaisen tiedonsiirtoon kuuluvan paketin tulee siis sisältää jokin tällainen tyyppitunniste, jotta saapuvaa dataa osataan tulkita oikein. Tämän sovelluksen kannalta kaikki olennaiset toimenpiteet saadaan aikaiseksi pakettityypeillä, jotka on esitetty alla taulukossa 6.1.

Taulukko 6.1. Anturiverkon pakettityypit.

Paketin tyyppi	tunniste (hex)	Pakettityypin toiminto
PACKETTYPE_ACK	0x30	Positiivinen kuittaus.
PACKETTYPE_NACK	0x31	Negatiivinen kuittaus.
PACKETTYPE_QUERY	0x32	Anturikysely.
PACKETTYPE_QUERYREPLY	0x33	Anturivastaus
PACKETTYPE_SETMODE	0x35	Anturisolmun parametrien asetus.
SETMODE pakettiin anturisolmut vastaavat joko positiivisella tai negatiivisella kuittauksella.		
PACKETTYPE_TICK	0x36	Mittauksen käynnistys.
PACKETTYPE_GETDATA	0x37	Datapyyntö.
PACKETTYPE_DATAAREPLY	0x38	Anturisolmun vastauspaketti

Tarkastellaan pakettityyppien merkitystä ja sisältöä seuraavassa lähemmin.

PACKETTYPE_ACK. Positiivinen kuittaus, jonka anturisolmu lähettää verkon masterille merkinä pyydetyn toiminnon onnistumisesta.

PACKETTYPE_NACK. Negatiivinen kuittaus, jonka anturisolmu vastaavasti lähettää tiedoksi pyydetyn toiminnon epäonnistumisesta. Tässä tapauksessa on usein kyse laittomista parametreista.

PACKETTYPE_QUERY. Tämä verkon masterin lähettämä paketti pyytää osoitettua anturia kertomaan sisältämänsä anturit. Broadcast -tyyppiseen lähetykseen anturit vastaavat osoitteidensa mukaisessa järjestyksessä.

PACKETTYPE_QUERYREPLY. Verkon masterille osoitettu vastaus _QUERY -pakettiin. Kiinteiden kenttien lisäksi sisältää yhden tavun dataa, joka kertoo käytettävissä olevat anturit seuraavan taulukon 6.2 bittien summana.

Taulukko 6.2. Bitti - anturi vastaavuudet.

0x01	Digitaalinen kiihtyvyys anturi SCA3000 asennettu
0x02	Analoginen kiihtyvyysanturi ADXL330 asennettu
0x04	Magneettianturi MS2100 asennettu
0x08	Kapasitanssianturi #1 eli AD7142 asennettu
0x10	Kapasitanssianturi #2 eli AD7142 asennettu

PACKETTYPE_SETMODE. Anturisolmun parametrien asetuspaketti. Datakenttä sisältää seuraavat parametritavut esitellyssä järjestyksessä.

SensorList -tavu, määrittää mitä solmun antureita halutaan käyttää. Sisältö koostuu kuten _QUERYREPLY -paketissäkin. Jos haluttua anturipiiriä ei löydy ohitetaan kyseinen anturi ilman virheilmoitusta, tällöin saadaan broadcast -tyyppinen verkon konfigurointi mahdolliseksi.

SampleRate -tavun arvon ollessa nolla pitää verkon masterin tuottaa näytteenottokello (TICK) mittauksen aloittamiseksi. Tällöin mittausdata on myös haettava manuaalisesti. Vaihtoehtoisesti arvo voidaan määrittää välille 1...100, jolloin anturiverkko lähettää päivitettyä dataa automaattisesti. Lähetystaajuus hertseinä on tällöin suoraan asetettu lukuarvo.

CompassResolution -tavu määrittää magneettianturin käyttämän resoluution. Arvot välillä 0...7 mahdollisia, jolloin nolla edustaa hitainta ja tarkinta mittausta ja arvo seitsemän nopeinta mutta epätarkinta.

CompassDivisor -tavu määrittää magneettianturin näytekellon esijakajan arvon. Koska magneettianturin toiminta on suhteellisen hidasta verrattuna muihin antureihin, on mittauksen aloitustaajuutta syytä hieman laskea. Esimerkkinä mainittakoon, että esijakajan arvolla 2, mittaus käynnistyy ainoastaan joka kolmannella näytekellolla. Esijakajan arvolla 0 toiminta on asynkronista eli mittausta suoritetaan niin nopeasti kuin mahdollista.

CapacitanceNofChannels -tavu määrittää kapasitanssianturissa käytettävien kanavien lukumäärän, joka voi olla maksimissaan 12.

CapacitanceOffset -tavu määrittää kapasitanssimittauksen offset -arvon pF:eina, jolloin arvot välillä 0...20 mahdollisia.

CapacitanceMode -tavu tulevaisuuden tarpeita varten. Asetettava nollassi.

CapacitanceDivisor -tavu määrittää esijakajan arvon kapasitanssimittauksen näytekellolle. Toiminta on identtinen magneettianturien näytekellon esijakajan kanssa.

PACKETTYPE_TICK. Jos **SampleRate** on _SETMODE paketissa määritelty nolllaksi, tulee isäntä -sovelluksen käynnistää mittaustapahtuma "manuaalisesti" tällä paketilla. Varsinaisella datakentällä ei tässä paketissa ole sisältöä.

PACKETTYPE_GETDATA. Käytettäessä manuaalista kellotusta (**SampleRate** = 0) tulee verkon isännän pyytää anturisolmuilta haluttuja mittausarvoja tällä erillisellä käskyllä.

PACKETTYPE_DATAREPLY. Anturisolmun lähettämän vastauspaketin tyyppi, joka sisältää anturisolmun mittaamat data -arvot. Paketin sisältö on samankaltainen _SETMODE -paketin kanssa, mutta tarkastellaan sisältöä hieman tarkemmin.

SensorList -tavu kertoo minkä antureiden dataa paketti sisältää.

NCapacitanceNodes -tavu kertoo kuinka monen kapasitanssikanavan dataa saapunut paketti sisältää.

Acceleration1 sisältää kolme 16 -bittistä lukua edustaen X-, Y- ja Z -akselien kiihtyvyysarvoja ensimmäiseltä kiihtyvyysanturilla (SCA3000). **Acceleration2** on niin ikään kolme kappaletta 16 -bittistä lukua edustaen mahdollisesti asennetun toisen kiihtyvyysanturin (ADXL330) arvoja.

Compass sisältää kaksi 16 -bittistä lukua edustaen magneettianturin (MS2100) mittausarvoja.

Capacitance1 sisältää niin monta 16 -bittistä kapasitanssi kanavan arvoa kuin mittauskanavia on määritetty käytettäväksi. **Capacitance2:n** arvot muodostuvat kuten Capacitance1:n jos toinen kapasitanssianturi on määritetty käytettäväksi.

6.2.2. Älypuhelimien ja keskusyksikön välinen yhteyskäytäntö

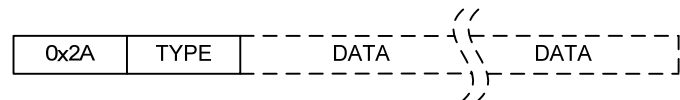
Keskusyksikön ja älypuhelimien välinen langaton tiedonsiirto toteutetaan Bluetooth -radiota ja eritoten tämän sarjaporttiprofiilia hyödyntäen. Käytössä olevalla Bluetooth -moduulilla on kaksi erillistä toimintatilaa; COMMAND- ja DATA -mode. Moduulin siirtyminen toimintatilasta toiseen asetetaan normaalisti toimimaan automaattisesti esimerkiksi yhteyden luonnin onnistuessa. Vaihtoehtoisesti tilan vaihto voidaan toteuttaa manuaalisesti HOST -prosessorin välityksellä. Keskusyksikön yhteyskontrolleri keskustelee COMMAND -tilassa Bluetooth -moduulin kanssa erityisen, moduulin valmistajan kehittämän, ASCII -pohjaisen iWRAP -rajapinnan kautta. Tämä rajapinta nopeuttaa huomattavasti ohjelmiston kehitystyötä, kunhan ohjelmoitsija on päässyt perille toiminnan peruseräpäätteistä ja käytännöistä [27].

Tässä sovelluksessa keskusyksikön ja puhelimen välisessä yhteydenpidossa keskusyksikkö toimii aina yhteyden master -osapuolena. Käytännössä tämä tarkoittaa, että keskusyksikkö hallitsee yhteyksiä kokonaisuudessaan ja muun muassa yhteydenottopyynnöt tapahtuvat aina keskusyksiköltä puhelimelle. Moduulin iWRAP -rajapinta tarjoaa käytettäväksi kymmenittäin ASCII -merkkipohjaisia käskyjä, joten

Bluetooth -radion osittain hyvinkin monimutkainen teknologia on saatu näin lähemmäs loppukäyttäjää. Karkeasti käskyt voidaan jaotella moduulin sisäinen toiminnan asetuksiin ja yhteysasetuksiin. Tarkempi rajapinnan käskyjen kuvaus selviää tarvittaessa rajapinnan dokumentaatiosta [26].

Keskusyksikön ollessa yhteydessä puhelimeen toimii Bluetooth -teknologia ”näkymättömänä” taustalla eli alemmalla protokollatasolla. Ennen kuin yhteys puhelimeen voidaan muodostaa, vaatii moduuli alustusrutiininsa, jotka HOST -prosessorin tulee tuottaa. Alustusrutiineiden sisältö pitää sovelluskohtaisesti selvittää hyödyntäen aiemmin mainittua iWRAP -rajapinnan dokumentaatiota.

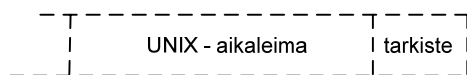
Edellisten perusteella voidaan ajatella keskusyksikön ja puhelimen välisen yhteyden toimivan ylemmällä protokollatasolla (sovellustaso) verrattuna tiedonsiirron mahdollistavaan RFCOMM -protokollaan. Myös tämän ylemmän yhteysprotokollan pakettien rakenteeksi haluttiin mahdollisimman optimoitu ratkaisu oman sovelluksen kannalta. Jälleen perusteena oli pienempi pakettien koko, jolloin moduulin ja koko järjestelmän virrankulutus pienenee. Lisäksi optimoidulla paketilla HOST -prosessorilta yhteyden aikana vaadittavat resurssit ovat nopeammin jälleen käytettävissä. Puhelimen ja keskusyksikön välillä kulkevat paketit ovat rakenteeltaan seuraavanlaisia.



Jokainen paketti siis sisältää vähintään kaksi tavun mittaista kenttää; vakiosuuruinen, heksadesimaalimuotoinen 0x2A sekä TYPE -kenttä. TYPE -kentän arvo määrittelee paketin (optionaalisen) loppuosan sisällön seuraavilla mahdollisilla arvoilla.

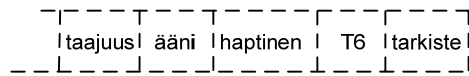
Tyypin **0x00** paketti on niin kutsuttu nolla -paketti, joka lopettaa yhteyden. Tällä paketilla ei ole optionaalista loppuosaa, ei edes tarkistettavaa.

Paketti tyypiltään **0x01** on puhelimen keskusyksikölle lähettämä aikaleimapaketti. Tässä pakettityypissä optionaalinen kenttä sisältää 32 -bittisen eli 4 tavua pitkän UNIX -aikaleimatiedon etumerkittömänä little-endian lukuna. Paketti lopetetaan tavun mittaiseen tarkistesummaan.

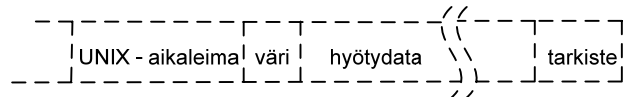


0x02 pakettitunnisteella lähetetään puhelimelta keskusyksikölle niin kutsuttu asetuspaketti. Taajuus on desimaaliarvona välillä 0...9, jolla määritellään

anturisolmujen ja keskusyksikön välinen lähetystaajuus. Muut arvot (ääni, haptinen ja T6) ovat joko tosia tai epätosia ja kertovat mitkä käytettävissä olevista varoituksista halutaan ottaa käyttöön. Myös tämä paketti loppuu tavun mittaiseen tarkistesummaan.



0x03 tunnisteella keskusyksikkö lähettää puhelimelle mahdollisesti tallettamansa tapahtumat. Näitä tapahtumia keskusyksikkö generoi jatkuvalla periaatteella siten, että raajan liikeradan ylitykset ja vaaratilanteet tallentuvat muistiin aikaleimalla varustettuna. Paketin UNIX -aikaleima on muodoltaan samanlainen kuin varsinaisessa 0x01 -aikaleimapaketissa. ”Väri” määrittelee minkä tyyppisen tapahtuman tiedot paketti sisältää (vaaratilanne vai rajanylitys). Hyötydatakentällä voidaan välittää anturitiedot tapahtumahetkellä ja paketti lopetetaan tarkistetavulla.



Kaikissa pakettityypeissä tarkistetavu on yksinkertaisesti paketin pituus tavuina, mukaan lukien itse tarkistetavu. Vaihtoehtoisesti tarkistetavuna voidaan käyttää kaikkien tavujen summaa, jolloin tarkistetavua ei lasketa mukaan.

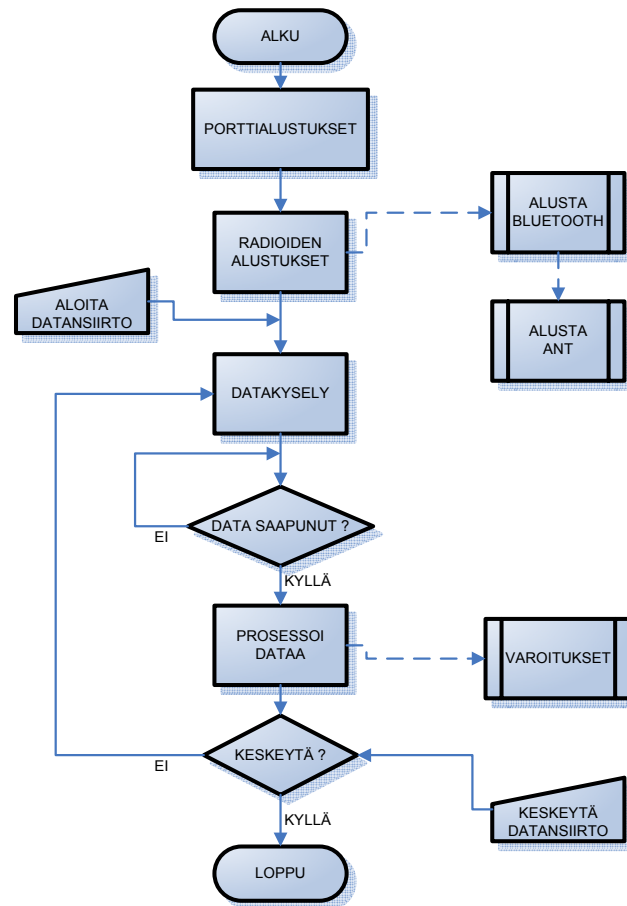
6.3. Ohjelmiston toteutus ja toimintaperiaate

Ohjelmiston varsinaisena kehitysympäristönä käytettiin mikrokontrollerivalmistaja Atmelin tuotteilleen tarjoamaa ilmaista kehitysympäristöä AvrStudio 4:ää. Tämä ympäristö sisältää muun muassa simulaattoriominaisuuden, jolla kehitettävien ohjelmistojen periaatteellista toimintaa voidaan simuloida PC ympäristössä jo varhaisessa kehitysvaiheessa. Kyseiseen ympäristöön on helposti integroitavissa ulkoinen kääntäjä, joka tässä tapauksessa oli niin ikään ilmainen WinAVR gcc. Vaihtoehtoja valinnalle olisivat olleet muun muassa kaupalliset IAR Embedded Workbench sekä CodeVisionAVR -kääntäjät. Yleisesti kaupallisten, jatkuvasti kehittyvien kääntäjien eduiksi katsotaan valmiiksi kirjoitetut ja optimoidut rutiinit tietyille oheislaitteille, hienot ja selkeät graafiset käyttöliittymät sekä mahdollisuus henkilökohtaiseen tukeen ongelmia kohdattaessa. Kaupallisten työkalujen avulla saavutettava lisähyöty katsottiin kuitenkin tällä kertaa niin marginaaliseksi, että päädyttiin ilmaiseen kehitysympäristöön. Pääsääntöisesti ohjelmisto on toteutettu C++ -ohjelmointikielellä ja toimintoja on tietyiltä osin optimoitu ja tehostettu assembly -kielestä poimituilla erilliskäskyillä.

Keskusyksikön ohjelmisto toimii niin sanotulla kiertokyselyperiaatteella (*engl. polling*). Käytännössä tämä tarkoittaa ohjelman toimintojen suoritusta silmukkamaisella, jatkuvalla rakenteella. Keskeytysperiaatetta, jossa ohjelmakoodin normaali suoritus keskeytyy väliaikaisesti jonkin tietyn toiminnon suorittamisen ajaksi, on hyödynnetty kuitenkin soveltuvien osin lisätoimintojen toteutuksessa. Ohjelmistokeskeytys on hardware -tasolla suhteellisen raskas toimenpide, sillä keskeytyksen saapuessa suorituksessa oleva ohjelmakoodi tulee pinota osoittimilla (*engl. stack pointer*) myöhempää suoritusta varten. Välttämällä keskeytyksien käyttöä voidaan mikrokontrollerin rajallisia resursseja ohjata vaativampaan raajan asennon laskentaan. Vielä merkittävämpi syy päätyä ohjelmistossa kiertokyselyperiaatteeseen on antureilta saatavan datan selkeämpi ajoitusten hallinta. Tällöin pääsilmukka määrittää milloin anturidataa halutaan saapuvaksi solmuilta ja voidaan lähes 100 % todennäköisyydellä olla varmoja saapuvien mitta -arvojen samanaikaisuudesta. Pääsilmukka jää siis tarvittaessa odottamaan saapuvaa dataa asetetun aikarajan (*engl. time-out*) puitteissa ja on näin ollen tietoinen mahdollisesti saapumatta jääneistä mitta -arvoista. Keskeytysperiaatetta käytettäessä tulisi ohjelmiston pitää yllä erillistä, resursseja kuluttavaa tilastoa saapuneiden mitta -arvojen ajankohdista.

Ohjelmistosta toteutettiin prototyyppivaiheessa kaksi hieman toiminnoiltaan poikkeavaa versiota, joista ainoastaan toinen mahdollisti yhteydenpidon matkaviestimeen Bluetooth -yhteyden välityksellä. Seuraavalla sivulla olevasta ohjelmiston vuokaaviosta kuvassa 6.2 nähdään periaatteellinen toiminta toteutetuissa ohjelmistoversiossa.

Laajuutensa vuoksi kokonaisen ohjelmakoodin esittäminen edes liitteenä ei ole mielekäästä. Sen sijaan liitteessä 3 on esitelty muutama ohjelmiston toiminnan kannalta olennainen, kommentoitu katkelma toteutetusta ohjelmakoodista. Ensimmäinen koodikatkelma on funktio, joka vastaan ottaa anturiverkolta saapuvia anturointidataa sisältäviä paketteja. Pakettien mahdolliset sisällöt on kuvattu aiemmin kappaleessa 6.2.1. Käytännössä funktio ottaa vastaan dataa tavu kerrallaan ja parserei sen keskusyksikön kannalta selkeämpään muotoon. Olennaisen tiedon datapaketista funktio palauttaa sitä kutsuneelle pääohjelmalle. Toinen esitellyistä koodikatkelmista esittää käytetylle WT12 Bluetooth -moduulille lähettävää asetussekvenssiä. Käytännössä tämä paketti sisältää kaiken tarvittavan informaation moduulin yhteyden muodostukseen aina laiteparien luomisesta sarjaliikenteen nopeuden asettamiseen.

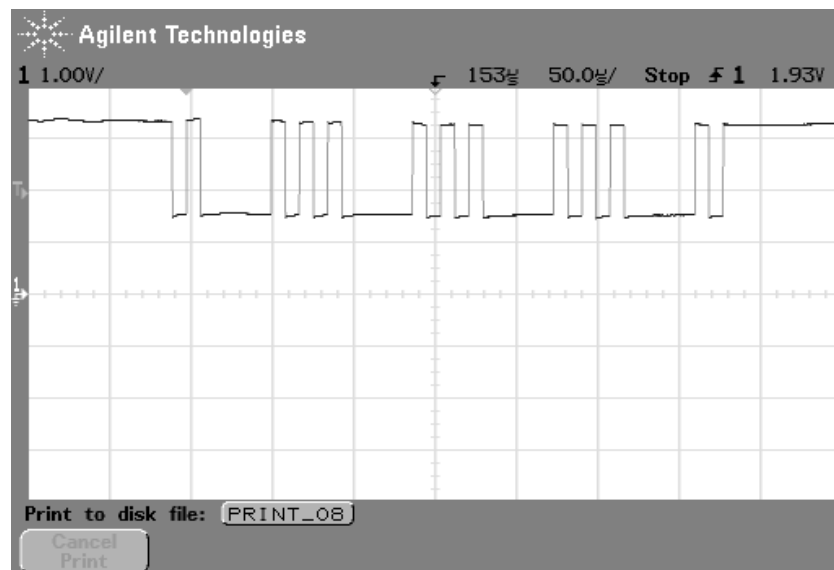


Kuva 6.2. Ohjelmiston vuokaavio.

Ohjelmiston suoritus alkaa sulautetuista järjestelmistä tutuilla toimenpiteillä, kuten porttialustuksilla ja globaalien muuttujien määrittelyillä. Radiopiirien alustukset on sijoitettu helppoa muokattavuutta ajatellen omiin aliohjelmiinsa. Tämä mahdollistaa nopean koodin sopeutuksen esimerkiksi siinä tilanteessa, että päädytäänkin syystä tai toisesta käyttämään vaihtoehtoisia radiotekniikoita. Tämän ratkaisun myötä tuli myös tarpeelliseksi kiinnittää erityistä huomiota moduulien julkisen rajapinnan suunnitteluun. Radioyhteyksien muodostumisen jälkeen käyttäjän on mahdollista aloittaa dataliikenne nappisyötteellä. Ohjelmisto siirtyy kiertokyselyperiaatetta noudattaen jatkuvaan silmukkaan, joka kysyy dataa anturiverkon solmuilta ja suorittaa samalla datan prosessointia. Ohjelmistoversiosta riippuen keskusyksikkö kykenee lähettämään joko asentodataa PC -sovellukselle tai vaihtoehtoisesti raja -arvo tapahtumia mobiililaitteelle. Käyttäjä voi missä tahansa vaiheessa keskeyttää datan siirron ulkoisella näppäinsyötteellä. Napin painallus rekisteröityy keskusyksikön muistiin ja datanvälitys keskeytyy päästäessä seuraavan kyselysilmukan loppuun. Keskeytyksen myötä keskusyksikkö siirtyy odottelutilaan kuitenkin sammuttamatta radiopiirejä. Näin ollen käyttäjä voi milloin tahansa aloittaa dataliikenteen alustamatta radioita uudestaan.

Dataliikenteen manuaalinen aloitus ja lopetusmahdollisuus katsottiin tarpeelliseksi prototyyppivaiheessa. Debuggauksen kannalta oli tärkeää, datan siirto saatiin alkamaan vasta kun Bluetooth -yhteys oli varmasti avautunut. Muussa tapauksessa Bluetooth -moduulin puskuri täyttyi nopeasti ja seurauksena oli useimmiten järjestelmän resetoituminen. Moduulin toimintaan tutustuesssa havaittiin myös useaan otteeseen, että ohjelmallisesti moduulilta saapunut tieto yhteyden avautumisesta oli usein vajavainen ja tästä syystä yhteyden tilan tarkkailu suoritettiin myöhemmissä versioissa omaa fyysistä linjaa käyttäen.

Toteutetut ohjelmistoversiot poikkeavat toisistaan ainoastaan käyttöliittymän osalta, eivätkä näin ollen toimi varsinaisen mittaustoiminnan kannalta eri tavalla. Bluetooth -mobiililaitetta hyödyntävässä ohjelmistoversiossa on mahdollista asettaa raajan asennon raja -arvot sekä järjestelmän kellonaika visuaalista ympäristöä käyttäen. Lisäksi mobiililaitteelle välitetään tieto mahdollisesti ylitetyistä raja -arvoista.



Kuva 6.3. Ongelmatilanne datansiirrossa.

Toteutetun koodin testausvaiheessa havaittiin ongelma eri anturisilloilta saapuvan datan tulkinnan luotettavuudessa. Kuva ongelmatilanteesta ennen korjausta oskilloskoopin ruudulta havainnollistettuna on nähtävissä yllä kuvassa 6.3. Ongelmaa jäljitettiin aina hardware -tasolla asti muun muassa oskilloskooppia apuna käyttäen. Tutkimusten perusteella todettiin ongelmien aiheutuvan mitä ilmeisimmin eroista Atmelin ja Texas Instrumentsin mikrokontrollereiden UART -sarjaliityntöjen sisäisten ylös- ja alaspäin suuntaisten viestien suuruuksissa. Käytännössä lähetyshetkellä passiivisen mikrokontrollerin datalinja oli riittävän voimakas häiritsemään lähettävän sillan toimesta tapahtuvaa datalinjan käsittelyä siinä määrin, että luotettava datan tulkinta estyi. Ongelmasta päästiin eroon poistamalla käytöstä lähetysten välillä kokonaan se anturisilta, joka ei kyseisellä ajanhetkellä ole mukana tiedonsiirrossa. Jotta korjaus

saatiin aikaiseksi, vaati se perehtymistä toisen projektiosapuolen tuottamaan ohjelmakoodiin. Lopulta kuitenkin anturisillat saatiin toimimaan yhteistyössä. Lopullista laiteversiota ajatellen anturisillan ja asennonlaskennan integrointi samalle mikrokontrollerille olisi poistanut kyseisen ongelman jo luonnostaan. Tällöin langalliselle ja langattomalle anturiverkolle olisi käytössä omat sarjamuotoiset liityntänsä.

7. KOTELOINTI JA KIINNITYS ASUSTEESEEN

7.1. Puettavuus

Koska järjestelmä sisältää päällepuettavan osion, tulee sen puettavuuteen ja sitä kautta käytettävyyteen kiinnittää erityistä huomiota. Puettavuuden käsitettä ja puettavan järjestelmän hyväksyttävyyttä käyttäjän kannalta on käsitelty monissa kirjallisissa teoksissa ja julkaisuissa [28]. Yleisellä tasolla puettavan laitteen sosiaalista hyväksyntää helpottavat kriteerit voidaan jakaa viiteen osa -alueeseen.

- **Puettavuus** eli miten helppo asuste on pukea ja pitää yllä. Miten hyvin asuste mukautuu toimintaan suorittaessamme jokapäiväisiä tehtäviämme?
- **Käytön helppous** määrittelee miten helppoa on käyttää laitetta ja sen tarjoamia palveluita. Kuinka paljon laitteen käyttö vie keskittymistämme siitä tehtävästä mitä olemme varsinaisesti tekemässä.
- **Esteettinen muotoilu.** Jos laite tai järjestelmä on tarkoitettu näkyvillä pidettäväksi, tulee sen vastata käyttäjän näkemystä visuaalisesti tyylikkäästä laitteesta.
- **Toiminnallisuus.** Tarjoaako laite käyttäjälleen toimintoja joista on todellista hyötyä tietyn toiminnon suorittamiseen. Tällöin järjestelmän ympäristö- ja tilannetietoisuus pitää olla riittävällä tasolla.
- **Hinta** on lähestulkoon aina ratkaisevassa roolissa suunniteltaessa laitetta kuluttajalle. Hinta ei kuitenkaan aina korreloi sen kanssa miten hienoa teknologiaa on toiminnan taustalla vaan useimmiten arvon määrittää se miten arvokkaaksi käyttäjä laitteen kokee.

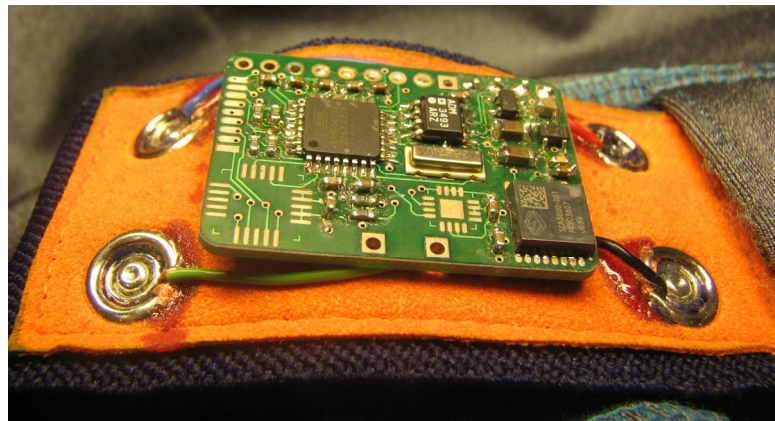
Puettavuutta suunniteltaessa voidaan lähtökohdaksi valita samoja edellä esiteltyjä kriteereitä. Lopullinen ratkaisu on tietysti aina sovelluskohtainen ja lähestulkoon aina kompromissi eri osa -alueiden kesken. [28]

HipGuard -housuissa liikkeellä lähdettiin ensisijaisesti toiminnallisuutta ja puettavuutta ajatellen. Edellä mainituista kriteereistä hinta ja esteettinen muotoilu olivat

toissijaisessa asemassa, koska oltiin kehittämässä vasta prototyyppijärjestelmää. Housujen materiaalivalinta ja leikkaus pyrittiin valitsemaan tekstiilialan ammattilaisia apuna käyttäen niin, että housuista tulisi mahdollisimman hyvin paikallaan pysyvät suurten lihasten liikkeistä huolimatta. Tämä ratkaisu vaikutti negatiivisesti puettavuuteen sillä kireähkö asuste on aina väljää hankalampi pukea. Ratkaisu koettiin kuitenkin välttämättömäksi anturisolmujen liikkeen minimoimiseksi.

7.2. Anturit ja keskusyksikkö asusteessa

Anturisolmuja asusteeseen liitettäessä haluttiin kehitysvaiheessa jättää mahdollisuus liikutella niitä lähestulkoon vapaasti radiaalisuunnassa jalkaan nähden. Housuihin ommeltiin anturisolmujen paikoille kanavat, joissa anturin kiinnitysalustana toimivaa leveää kuminauhaa oli mahdollista käännellä jalan ympäri. Tämä ratkaisu mahdollisti optimaalisen anturisolmun sijainnin etsimisen järjestelmää testattaessa. Anturisolmujen tuli lisäksi vaatimusten mukaisesti olla helposti irrotettavissa asusteesta pesua varten. Langattoman anturisolmun irrotus onnistui tarranauhan avaamisella, mutta langallisen solmun kiinnittämiseen pyrittiin keksimään uusi ja toimiva ratkaisu.



Kuva 7.1. Anturisolmu nepparikiinnityksineen.

Johdollisen anturisolmun tapauksessa päädyttiin hyödyntämään kiinnityksessä tekstiiliteollisuudessa yleisesti käytössä olevia metallisia neppareita. Ratkaisua havainnollistaa yllä oleva kuva 7.1. Neppareiden toiset puolet on kiinteästi asennettu housujen koteloituihin kuminauhoihin. Toiset puolet ovat vastaavasti kiinni pienessä kankaanpalassa (kuvassa oranssi kangas), joka toimii myös piirilevyn liitosalustana. Kontaktit piirilevyn ja neppareiden välillä on tässä toteutettu tavallisella johdolla. Toinen vaihtoehto olisi käyttää taipuisaa piirilevyä kankaan tilalla. Lopullinen versio tultaisiin valamaan polyuretaanivaluun mekaanisilta vaurioilta suojautumiseksi. Neppareihin perustuvan ratkaisun ehdottomasti paras puoli on, että sekä fyysinen että sähköinen kontakti saadaan aikaan samalla toimenpiteellä. Anturisolmujen kestävyydestä ei tällä kiinnitysmenetelmällä ollut mahdollista aikataulujen puitteissa

suorittaa testejä. Selvä tarve elektroniikan suojaamiselle esimerkiksi polyuretaanivalun muodossa oli kuitenkin nähtävissä jo käytön alkuvaiheessa.



Kuva 7.2. Anturisolmut ja keskusyksikkö housuihin sijoitettuna.

Yllä olevassa kuvassa 7.2 on nähtävissä anturisolmujen ja itse keskusyksikön kokonaisvaltainen sijoittelu housuihin. Kuvassa on nähtävissä housuihin johdotusta varten ommellut johtokourut sinisten neulosten väliin jäävänä alueena. Anturisolmut kuminauhakouruineen ovat niin ikään nähtävillä nilkan ja polven yläpuolella. Järjestelmän keskusyksikkö näkyy vyötäröllä harmaaseen laatikkoon koteloituna. Tämä prototyypin kotelo on valmistettu jyrsimällä tavallisesta PVC -muovista. Kotelo on niin ikään kiinnitetty asusteeseen edellä kuvattua nepparimenetelmää käyttäen.

7.3. Joustavat kotelointimateriaalit

Perinteisesti elektroniikkaa on pyritty suojaamaan kosteudelta ja jossain määrin myös mekaanisilta rasituksilta erityyppisiä epoksihartseja käyttäen. Jäykkyytensä vuoksi nämä materiaalit ovat kuitenkin soveltumattomia käytettäväksi taipuisien piirilevyjen suojaamiseen. Näin ollen taipuisat piirilevyt vaativat suojaukseensa aivan uudentyypisiä, joustavia materiaaleja.

Elektroniikan suojaaminen sinänsä asettaa jo tiukat vaatimukset suojaamiseen käytettävälle materiaaleille. Näitä vaadittavia ominaisuuksia ovat muun muassa riittävä sähköinen ja kemiallinen eristyskyky, pieni lämpötilariippuvuus dimensioiden

säilyttämiseksi vaihtelevissa olosuhteissa sekä suojausmateriaalin soveltuvuus käytettäväksi sähköisten komponenttiliitosten kanssa. [29]

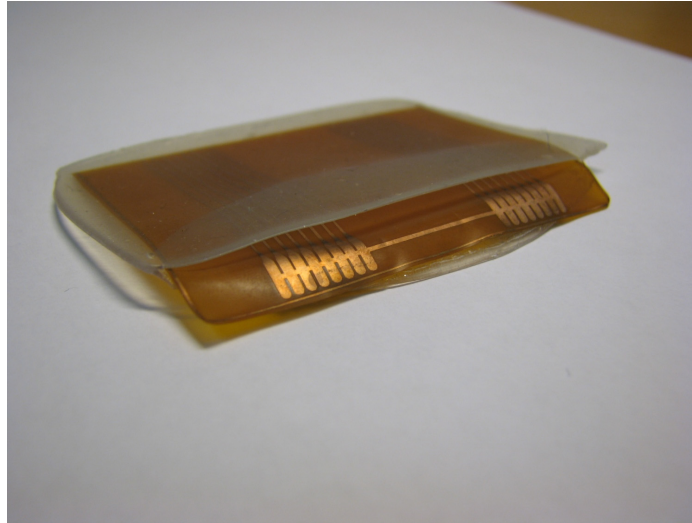
Käytännössä viimeinen edellä mainituista ominaisuuksista viittaa elektronisten komponenttien lämpötilakestävyyteen, sillä useat integroidut, erityisesti mikro - elektromekaaniset, piirit eivät kestä joidenkin suojamateriaalien kovettamiseen vaadittavia lämpötiloja. Kiteillä ja useilla akkutyypeillä suurin mahdollinen lämpötilansieto saattaa jäädä jopa vain 45 °C:een, vaikka käytännön testit ovatkin osoittaneet huomattavasti suurempia lämpötilankestoja. Useimmat suojaukseen käytettävät materiaalit kovettuvat myös normaaleissa huonelämpötiloissa, mutta lopputuloksen laatua voidaan parantaa lämpötilaa kasvattamalla. Samoin erityisesti teollisessa, suuriin tuotantoeriin keskittyvässä valmistuksessa merkityksellistä kovettumisaikaa voidaan lyhentää lämmön avulla.

Nykyaikainen tapa suojata taipuisia piirilevyjä on hyödyntää erilaisia elastomeereja. Näiden materiaalien hyödyntäminen on edelleen aktiivisena tutkimusalueena elektroniikan alalla ja kaikin puolin ylivertaista materiaalia ei ole vielä kehitetty. Teoriassa elastomeerit ovat yksi lupaavimmista materiaaleista, mitä tulee sähköisiin ominaisuuksiin, joustavuuteen ja kustannuksiin. Karkeasti amorfiset eli kiderakenteettomat elastomeerit voidaan jakaa termoelasteihin ja kumeihin. Käytännön erona näille elastomeeriryhmille on kimmoisuuden säilyminen. Kumi vaatii lämmön avulla tapahtuvan vulkanoinnin kimmoisuutensa säilyttämiseksi eikä salli uudelleenprosessointia [30]. Termoelastit sitä vastoin ovat luonteeltaan kestopuoveja ja näin ollen uudelleen muovattavissa. Elastomeerituotteiden valmistaminen poikkeaa merkittävästi muovituotteiden valmistuksesta. Ominaisuuksista johtuen mahdollisia valmistusmenetelmiä ovat lämpömuovaus, muottimenetelmät ja erityisesti elektroniikan suojaamisen kannalta tärkeät valaminen ja pinnoittaminen.

Taipuisassa elektroniikassa myös pinnoitteen tulee luonnollisesti olla taipuisaa, jossain määrin jopa joustavaa sekä kyetä myötäilemään liitosalustan liikkeitä. Toisaalta suojamateriaalin tehtävänä on myös tukea levyä ja estää liiallista taipumista ja tätä kautta hajoamista.

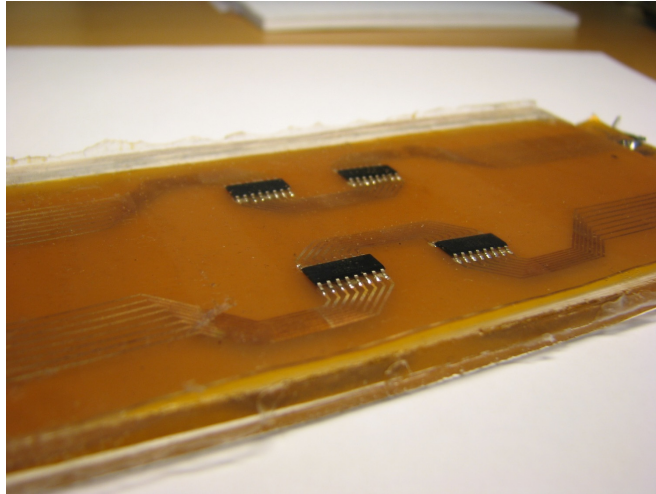
Osaa elastomeereista käytetään myös liimoina esimerkiksi liitettäessä elektroniikkaa suoraan tekstiiliin, kuten tullaan tekemään HipGuard -järjestelmän lopullisessa versiossa. Tällöin mukaan tulee erityisesti materiaalitekniikasta tuttu termi, adheesio joka kuvastaa kahden aineen välistä kiinnittymistä. Vaihtelevien piirilevy materiaalivalintojen vuoksi on adheesio usein tutkittava jopa tapauskohtaisesti ja tarvittaessa käytettävä erillistä liimaa kiinnityksen parantamiseksi. Toisaalta tulee myös tapauskohtaisesti miettiä adheesio tarve, sillä suinkaan aina ei ole paras vaihtoehto että elektroniikka on tiukasti kiinni pinnoitteessa. Heikko adheesio mahdollistaa elektroniikan irrottamisen pinnoitteesta esimerkiksi korjaus tai

muutostarkoituksessa. Suurempia tuotantoeriä ajatellen tulee nykypäivänä jo lakisääteisestikin huolehtia tuotteen huollettavuudesta sekä kierrätettävyydestä ja voimakas adheesio luonnollisesti vaikeuttaa tuotteen jatkokäsittelyä [31]. Alla olevasta kuvasta 7.3 on helposti nähtävissä heikon adheesion vaikutus pinnoitteen kiinnittymiseen.



Kuva 7.3. Esimerkki heikosta adheesiosta.

Rinnakkaisten projektien tutkimustuloksien perusteella toimivimmiksi suojamateriaaleiksi elastomeereista ovat osoittautuneet polyuretaanikumi ja silikonikumi sekä vaativamman teollisen prosessin mahdollistama solustettu polyuretaani [30]. Tavallisesti suojaukseen käytettävät polyuretaanikumit ovat kaksikomponenttisia, huoneenlämmössä vulkanoituvia materiaaleja. Suojaukseen hyvin soveltuvat materiaalit ovat kovuudeltaan luokkaa 50 – 60 ShoreA -asteikolla mitattuna. Polyuretaanikumien kiinnittyvyys piirilevymateriaaliin on tavallisesti todella hyvä ja muodostuva pinta on mahdollista ihokosketusta ajatellen sileä ja miellyttävän tuntuinen. Silikonikumit ovat niin ikään 2 -komponenttisia ja tarjoavat hieman polyuretaaneja paremman taipuisuuden, tosin kestävyys kustoannuksella. Silikonikumin valinnassa tulee käyttää erityistä huolellisuutta, sillä joissakin tapauksissa niiden todella heikko adheesio ja repeytymisherkkyys estävät niiden käytön taipuisan elektronikan suojauksessa. Edelleen polyuretaanin eduksi voidaan katsoa parempi mekaaninen kestävyys esimerkiksi pesutestien yhteydessä, vaikka silikonikumi yleisesti imee itseensä vähemmän kosteutta paremman solurakenteensa ansiosta. Seuraavassa kuvassa 7.4 on esimerkki silikonikumiin valetusta elektronikasta. Kyseisellä levytyypillä on tehty useita piirilevyn mekaanista kestävyyttä mittaavia testejä.



Kuva 7.4. Elektroniikkaa polyuretaanivalussa.

Yllä esitellyllä periaatteella on myös valmistettu polyuretaani- ja silikonikumiin valettuja akkuyksiköitä. Nämä käytännön testit ovat osoittaneet, että herkkienkin komponenttien suojaaminen elastomeerimateriaaleilla on mahdollista. [23] [30]

Eristävänä materiaalina suojamateriaaleilla on tietty vaikutus järjestelmässä käytettävien suuritaajuuksisten radioiden toimintaan, mutta kattavia tutkimuksia aihepiiristä on varsin rajallisesti saatavilla. Aihepiiriin perehtyminen olisi henkilökohtaisen kiinnostuksen sekä Kankaanpään Yksikön tutkimusalueena varsin mielenkiintoinen. Jälleen kerran projektin aikarajojen puitteissa ei ollut mahdollista paneutua aihepiiriin tarkemmin vaan jättää asia mahdollisen jatkokehittelyn piiriin. Järjestelmän testausvaiheessa erilaisilla eristeillä ja suojilla ei tuntunut olevan merkittävää vaikutusta radioiden suorituskykyyn, vaikka varsinaista polyuretaanivalua ei käytettykään.

8. JÄRJESTELMÄN TESTAUS JA TULOKSET

”Jos laitetta on ylipäättään mahdollista käyttää väärin, niin joku jossakin vaiheessa tulee niin tekemään”.

Testaus on välttämätön vaihe tuotteen laadun varmistamisessa. Puutteelliset testausvaiheet saattavat näkyä laitteen tai järjestelmän toiminnasta vastuussa olevalle taholle monenlaisina ei-toivottuina ilmiöinä. Käyttäjän kannalta taas ohjelmistovirheet tulevat esille epämiellyttävänä ja tietyissä tapauksissa jopa vaarallisena toimintana. Sulautetuissa järjestelmissä yleisin virhetilanteiden aiheuttaja onkin juuri laitteen ohjelmisto. Niiden lisäksi laitteen puolijohdepiirit aiheuttavat satunnaisesti omat virhetilanteensa esimerkiksi altistuessaan ulkoisille häiriöille kuten suuritaajuisille ja -tasoisille radiotaajuuksille tai staattisille sähkönpurkauksille. Näihin verrattuna ohjelmistovirheet ovat kuitenkin selvästi systemaattisempia ja siten mahdollista karsia järjestelmää testaamalla. [32]

Varsinkin kuluttajille suunnatuissa massatuotteissa ohjelmiston päivittäminen on kallis ja aikaa vievä operaatio, joka harvoin kohentaa tuotteen arvoa käyttäjän näkökulmasta. Erikseen on tietysti mainittava päivitykset joiden tarkoituksena on lisätä tuotteen ominaisuuksia eikä niinkään korjata olemassa olevia ohjelmistovirheitä. Käytännön esimerkit ovat useasti osoittaneet, että virheestä aiheutuvat kustannukset karkeasti arvioiden moninkertaistuvat siirryttäessä tuotteen elinkaareissa vaiheesta toiseen. Arvioiden mukaan pelkästään Yhdysvaltojen kansantaloudessa ohjelmistovirheiden kustannukset olivat vuonna 2002 60 miljardin dollarin suuruusluokkaa ja säästöpotentiaaliksi arvioitiin 22 mrd. dollaria [33]. Viime vuosina merkittävästi lisääntyneen ohjelmistomäärän myötä tilanne ei ole ainakaan parantunut. Kaavio 8.1 havainnollistaa kyseistä elinkaariketjua vesiputousmallissa usein käytetyillä vaiheilla esitettynä [34]. Määrittelynä tuotteen elinkaarella tarkoitetaan aikaa, joka kuluu ohjelmiston kehittämisen aloittamisesta sen poistamiseen käytöstä. Vastaavasti vaihejakomallilla tarkoitetaan tapaa, jolla ohjelmiston kehitystyö tai koko elinkaari jaetaan vaiheisiin [35].



Kaavio 8.1. Ohjelmiston elinkaari ja ohjelmistovirheen kustannukset.

Sytä puutteelliseen ohjelmistotestaukseen löytyy useita. Vähäisimpänä eivät varmastikaan ole testauksen tuotekehitystä hidastava vaikutus sekä testauksen vaatima suuri manuaalisen työn määrä, joka johtaa nopeasti kustannusten nousuun.

8.1. Käyttäjäkokemusperustainen testaus

Käyttäjäkokemusperusteisen (*engl. User Experience*) testaamisen tarkoitus on selvittää miten hyvin toteutunut järjestelmä on vastaa käyttäjän odotuksia. Mahdollisimman täydellisesti toteutettuna käyttäjäkokemus testaa tietyn palvelun tai tuotteen kuluttamiseen ja käyttämiseen liittyvää kokonaisvaltaista elämystä. Käsite siis kattaa käytettävyyden ja käyttöliittymän ohella joukon tunteita ja elämyksiä, joista osa saattaa ilmetä vasta pitkienkin aikojen kuluessa. Tästä esimerkkinä mainittakoon järjestelmän jatkuvasta käytöstä mahdollisesti aiheutuva henkinen tai fyysinen riippuvuus. Näiden mahdollisten pitkäaikaisten vaikutusten arviointia ei valitettavasti ole mahdollista projektin aikataulun puitteissa toteuttaa.

Käyttäjäkokemus on aina yksilöllinen (*engl. subjective*), jolloin kattavan testiotoksen saamiseksi testihenkilöitä tulisi siis useita. Muutenkin testausta suorittava henkilö tulisi olla projektin ulkopuolinen taho riippumattomien testaustulosten saavuttamiseksi. Tällä kertaa jouduttiin tyytymään kuitenkin yhden testihenkilön käyttöön, jonka kokoon toteutetut housut oli valmistettu. Lisäksi kyseinen koehenkilö oli työskennellyt kehitystyössä itse projektissa. Karkeasti käyttäjätestaus jaettiin kolmeen kokonaisuuteen:

- Laitteen käyttöönotto eli päälle pukeminen.
- Laitteen varsinainen käyttötilanne.
- Laitteen vaatimat huolto- ja ylläpitotoimenpiteet.

Asusteen päälle pukeminen sujui suorituissa testeissä suhteellisen vaivattomasti, huolimatta raajoja tiukasti mukailevasta leikkauksesta. Pukemisvaiheessa anturisolmut oli valmiiksi kiinnitetty asusteeseen aiemmin kappaleessa 7.2 esitellyllä neppari kiinnityksellä. Koska antureita ei ollut fyysisesti suojattu mitenkään, tuli pukemisvaiheessa kuitenkin noudattaa erityistä varovaisuutta. Selvä tarve solmujen suojaamiselle oli siis olemassa. Varsinaisessa käyttötilanteessa eniten käyttäjää häiritsevä tekijä on keskusyksikön, suurelta osin koteloinnista johtuva, paino. Vyötärölle kuminauhaan asennettuna laite pyrki kääntymään alaspäin oman massansa vaikutuksesta. Tästä ilmiöstä tultaisiin pääsemään ainakin osittain eroon vaihtamalla keskusyksikön suojaus kovasta PVC –koteloinnista taipuisaan valukotelointiin. Tällöin säästyneen painon lisäksi keskusyksikön painopiste siirtyy lähemmäs kehoa, joka vähentää kääntymistä. Järjestelmän jokapäiväiset huoltotoimenpiteet todettiin riittävän vähän käyttäjää rasittavaksi. Ainut päivittäin pakollisena suoritettava toimenpide on akun vaihto, joka vaatii käytännössä keskusyksikön irrotuksen ja kiinnityksen. Harvemmin suoritettava housujen pesu vaatii edellisen lisäksi anturisolmujen irrotuksen, joka tosin on saatu nopeaksi neppari kiinnityksen myötä.

Bluetooth -radiolinkin suorituskyky niin siirtokapasiteetin kuin erityisen tärkeän vasteviiveenkin kannalta oli käyttäjän mielestä riittävä. Toisin sanoen järjestelmä reagoi riittävän nopeasti tilanteisiin varoituksella jos oltiin lähestymässä raajan kiellettyä asentoa. Ongelmia ilmeni kuitenkin käyttäjälle näkymättömän ANT –radiolinkin toiminnassa. Radiolinkin teho ei yksinkertaisesti riittänyt luotettavan datansiirron aikaansaamiseksi ja tästä syystä langaton anturiverkko jätettiin käyttämättömäksi viimeisessä versiossa.

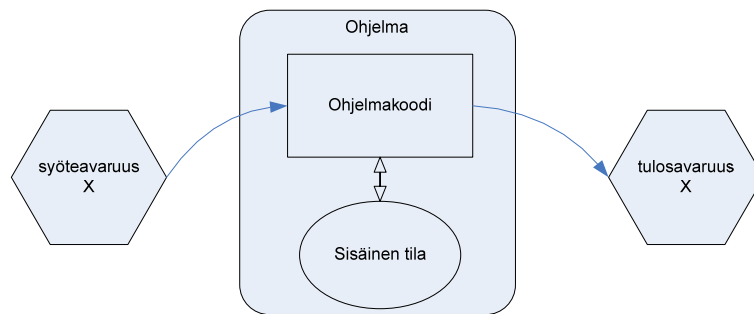
8.2. Tekninen testaus

Järjestelmän tekninen testaus käsittää niin aiemmin esitellyn ohjelmiston testausta kuin teknisten suureiden kuten virrankulutuksen ja datansiirtokapasiteetin mittausta ja arviointia. Niin ikään järjestelmän vasteviive on toiminnan kannalta olennainen osa - alue, mutta tämän mittaamiseen käytettävissä olevat laitteet olivat puutteelliset. Näin ollen suurin osa viiveen mittauksesta suoritettiin käyttäjäkokemuspohjaisena, kuten edellisessä kappaleessa 8.1 esitettiin.

8.2.1. Ohjelmiston tekninen testaus

Ohjelmiston testaus on systemaattista virheiden etsimistä, ei sattumanvaraista ohjelman kokeilua. Ohjelmaa tai ohjelman yksittäistä osaa testattaessa tulee ensimmäisenä määritellä syöteavaruus. Ongelmaksi tässä usein muodostuu

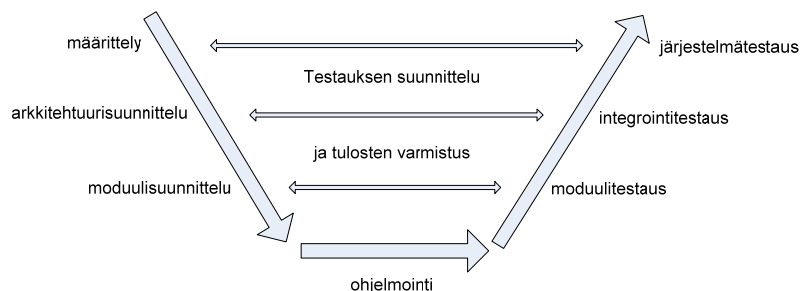
syöteavaruuden koko, joka saattaa jo pienehkölläkin ohjelmalla muodostua niin suureksi, ettei täydellinen testaaminen ole käytännössä täysin mahdollista. Täten toteutetun järjestelmän kohdalla keskityttiin tiedossa oleviin erityistapauksiin, ja syöteavaruus muodostettiin vastaamaan mahdollisimman hyvin käytännön tilannetta. *Lausekattavuus* mittarina määrittelee miten suuren osan ohjelmamoduulin lauseista yksikkötestaus kattaa. Tälläkin kertaa pyrkimyksenä oli hyvin valitun syöteavaruuden avulla saavuttaa moduulitason testauksessa 100 % lausekattavuus. Käytännössä hyvin valittu syöteavaruus sisältää luvallisia ja luvattomia syötteitä. Alla oleva kuva 8.1 havainnollistaa ohjelmiston testausta syöte- ja tulosavaruuksien suhteen. [36]



Kuva 8.1. Ohjelmiston toiminta testausta suunniteltaessa.

Kuten kuva edellä osoittaa tulee syöteavaruuden lisäksi määritellä tulosavaruus, joten testaajalla pitää olla tieto miten syötteet kuvautuvat tulosavaruuteen. Tulosavaruuden sisältö määräytyy useimmiten jo ohjelmiston määrittelyvaiheessa. Tavallisesti tulosavaruus on syöteavaruutta huomattavasti pienempi. Tulosavaruus voi olla esimerkiksi yksinkertainen totuusarvo vaikka syötteenä olisi sattumanvarainen ja mahdollisesti suurikin numeroarvojoukko.

Tulosavaruuden kuvautuminen oikeaksi käytetyllä syöteavaruudella ei kuitenkaan kerro koko totuutta ohjelman oikeasta toiminnasta. Edellä mainittujen lisäksi tulee huomioida ohjelman sisäinen tila. Toisin sanoen sisäisen tilan muutokset saattavat aiheuttaa satunnaisia muutoksia tulosavaruuteen, jos testaus ei ole kyllin kattava. Tällainen virhetilanne saattaa aiheutua vaikkapa ohjelman sisäisen tilakoneen testaamatta jääneistä tiloista.



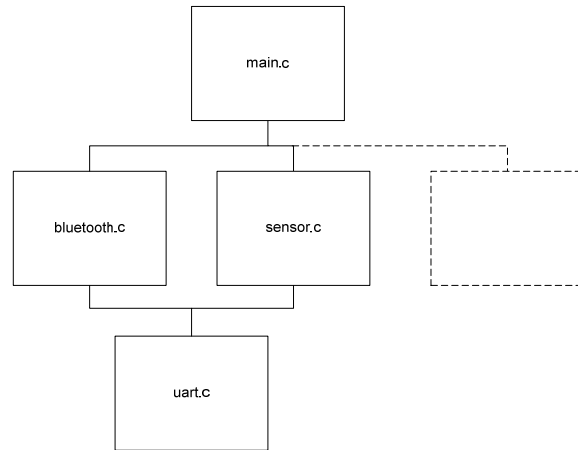
Kuva 8.2. Testauksen ja suunnittelun V-malli

Testaustasojen sisällön suunnittelun liittymistä eri suunnitteluvaiheisiin voidaan havainnollisesti kuvata erityisellä kuvan 8.2 V -malli kaaviolla. Toteutetun ohjelmiston testaus suoritettiin soveltuvien osien mahdollisimman aikaisessa ohjelmiston toteutusvaiheessa eri testaustasojen jakoperiaatetta mukaillen. [37]

- Testaus *moduulitasolla* käsitti modulaarisella periaatteella toteutetun ohjelmiston jokaisen loogisen osakokonaisuuden testaamisen erikseen. Täten voitiin mahdollisimman suurella todennäköisyydellä varmistua, että testattu moduuli ei väärällä toiminnallaan aiheuttaisi jatkossa ongelmia ohjelmistokomponentteja integroitaessa. Nyrkkisääntönä voidaan kokemuksen perusteella pitää, että koodirivejä testataan tässä vaiheessa 100 - 1000, riippuen tietysti ohjelmiston koosta. Esimerkkinä HipGuard -järjestelmän ohjelmistossa mainittakoon sarjaliittymän fyysisellä tasolla tapahtuva käsittely, jonka varaan koko sarjamuotoinen tiedonsiirto järjestelmän sisällä nojautuu. Moduulin testausta varten kirjoitettiin pienimuotoinen, testattavaa moduulia hyödyntävä, pääohjelma jolla erilaisia syötteitä vaihtelevilla viiveillä annettiin sarjaliittymämoduulin välitettäväksi. Erityisesti kiinnitettiin huomiota puskurimuistien riittävyyteen, jotta esimerkiksi informaation ylivuoto (*engl. data overflow*) voitiin välttää. Havaitut virheet moduuleissa korjattiin, jonka jälkeen oltiin valmiita siirtymään seuraavalla testaustasolle.
- Vaikka tietyn toiminnon toteuttava moduuli toimisikin yksin oikein saattaa moduulien lisääminen ohjelmistoon aiheuttaa odottamattomia vaikutuksia. *Integraatiotestauksen* tavoitteena pääasiallisesti testata ohjelmistomoduulien muille moduuleille tarjoamina rajapintoja. Tämä edellyttää, että moduulin sisäinen toiminta on tässä vaiheessa olla tarpeeksi pitkälle testattua. Tämä siksi, että integraatiotestaus on hyvin harvoin helposti täysin eriytettävissä moduulitestauksesta. Integraatiotestauksessa kirjoitettavat testiajurit (*engl. driver*) korvaavat moduulihierarkiassa testattavan moduulin yläpuolelle sijoittuvat eli testattavaa moduulia käyttävät tai kutsuvat ohjelmisto komponentit. Niin kutsutut testityngät (*engl. stub*) ovat taas moduuleita (tai muita vastaavia komponentteja), jotka ovat korvaamassa testattavien moduulien kutsumia komponentteja.

Integraatiotestauksen kolme eniten käytettyä lähestymistapaa ovat jäsentävä (*engl. top-down*), kokoava (*engl. bottom-up*) ja niin kutsuttu ”sandwich” -menetelmä, joka on yhdistelmä edellisistä. Omassa toteutetussa järjestelmässä integrointia suoritettiin soveltuvien osien juuri molemmista suunnista. Kuten moduulitestauksessakin testitapaukset pyrittiin valitsemaan siten, että mahdollisimman paljon virheitä saataisiin esille. Toteutetun

ohjelmiston moduulihierarkiaa ohjelmiston sisällä kuvaa alla oleva kuva 8.3. Oman ohjelmistomme moduuleita testattiin hierarkisesti pääosin kokoavalla periaatteella. Esimerkki muutaman toteutetun ohjelmamoduulin riippuvuussuhteista on nähtävissä kuvassa.



Kuva 8.3. Osa moduulihierarkiasta toteutetussa ohjelmistossa

Puhtaasta ohjelmistokehityksestä poiketen sulautettujen järjestelmien ohjelmistoissa tulee usein ottaa huomioon liityntä fyysiselle tasolle, esimerkiksi järjestelmän I/O -linjoihin. Omassa ohjelmistokoodissa mainittu liityntä tapahtuu yllä esitetyssä hierarkiassa alimmalla tasolla, UART -lohkoa käyttävässä moduulissa. Näin ollen kyseinen moduuli on luonnollinen valinta ensimmäiseksi testauskohteeksi. Tämä siksi, että kyseisen moduulin toimiessa luotettavasti muun ohjelmiston testaus helpottuisi olennaisesti. Sekä Bluetooth- että anturiverkkoa operoivat moduulit ovat kuvan mukaisesti samanarvoisia. Tästä syystä kolmea alimmaista moduulia testattiin ja kokoavalla periaatteella. Pääohjelmamoduulin (`main.c`) testaus suoritettiin sitä vastoin testisyötteitä antaen ennen varsinaista moduulien integrointia.

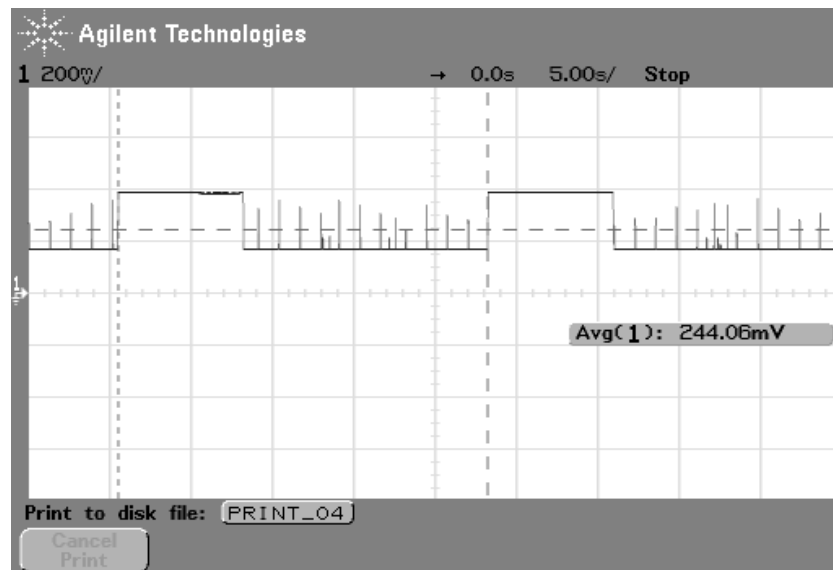
- Koko toteutettu järjestelmä on tarkastelun kohteena *järjestelmätestauksessa*. Kyseinen testi kokoaa jo kattavasti koko ohjelmiston ominaisuudet mukaan testiin. Järjestelmätestaus on usein viimeinen suoritettavista testausvaiheista. Joskus järjestelmätestauksen yhteyteen, varsinkin tuotetta tehdessä, voidaan ajallisesti viimeisenä toteuttaa *hyväksymistestaus*. Joskus hyväksymistestaus vaatii laajoja toimenpiteitä, jolloin kannattaa se kuitenkin eriyttää omaksi kokonaisuudekseen. Tämän mahdollisen viimeisen testivaiheen tarkoituksena on osoittaa asiakkaalle tai muulle vastaavalle loppukäyttäjälle, että järjestelmä kykenee suoriutumaan sille määritetyistä tehtävistä. Tällä kertaa erillistä hyväksymistestausta ei katsottu tarpeelliseksi toteuttaa, koska kyseessä oli ainoastaan prototyyppijärjestelmä. [36] [37]

8.2.2. Virrankulutuksen mittaus

Riittävän pieni virrankulutus mahdollistaa toteutetun järjestelmän soveltuvuuden pitkäaikaismittauksiin. Vaikka järjestelmän tehonsyöttö on toteutettu verrattain kohtuullisen suurikapasiteetisella akulla, tuli jo suunnitteluvaiheessa kiinnittää huomiota yksittäisten komponenttien tehonkulutuksen optimointiin. Käytetyistä komponenteista Bluetooth -moduuli ja varsinainen mikrokontrolleri tarjosivat valmiita virransäästöominaisuuksia, joita oltiin halukkaita testaamaan.

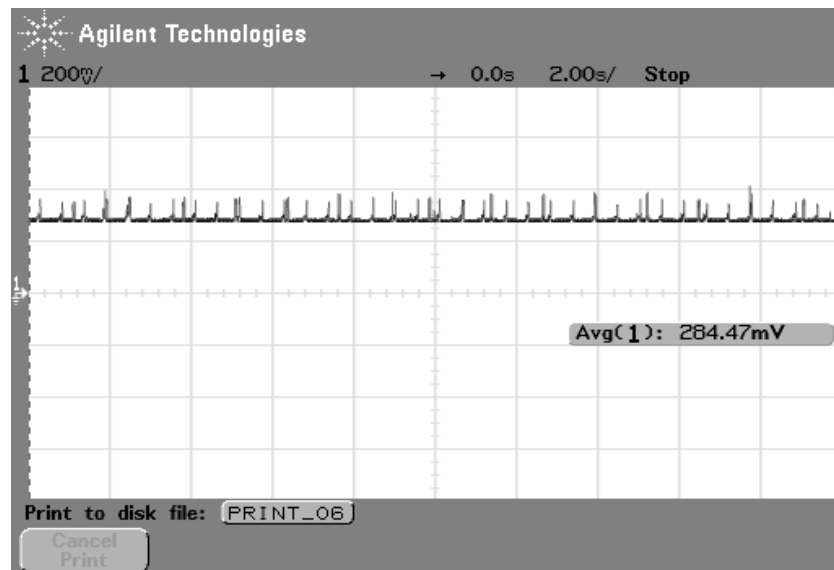
Laboratoriokäyttöön tarkoitettu yleismittari tehollisarvomittauksineen (niin kutsuttu trueRMS) ja keskiarvoistuksineen antaa jo jonkinlaisen käsityksen järjestelmän kokonaisvirrankulutuksesta. Valitettavasti tämä menetelmä ei juurikaan anna informaatiota siitä miten virrankulutus vaihtelee järjestelmän eri toimintatilojen myötä. Tässä vaiheessa käyttöön voidaan ottaa oskilloskooppi ja resistanssiarvoltaan riittävän pieni vastus joka toimii niin kutsuttuna virranmittausvastuksena. Tällöin virranmittausvastus kytketään siten, että kaikki järjestelmän kuluttama virta kulkee mittausvastuksen lävitse. Kaksikanavaisen oskilloskoopin differentiaalimittauksella mitataan vastuksen yli vallitseva jännite, josta voidaan edelleen helposti laskea hetkellinen virta tunnettaessa resistanssiarvo. Vaikka menetelmä ei ole aivan eksakti antaa se silti riittävästi tietoa järjestelmän käyttäytymisestä. Mittauksissa käytetty oskilloskooppi oli kaksikanavainen Agilentin valmistama malli 54622A.

Alusta alkaen oli selvää, että suurin yksittäinen virtaa kuluttava kokonaisuus oli toteutetussa järjestelmässä Bluetooth -moduuli [19]. Kyseisen moduulin virrankulutus aiheuttaakin suuria vaihteluita virrankäyttöprofiiliin kuten seuraavasta oskilloskoopin kuvasta 8.4 on nähtävissä.



Kuva 8.4. Virrankulutusprofiili Bluetooth yhteydettömänä.

Kuvasta on selkeästi nähtävissä virrankulutuksen syklittäinen muutos ajan funktiona. Virrankulutustason hetkellinen kasvu johtuu Bluetooth -moduulin yhteydenmuodostustilasta (*engl. connect*) eli kuvassa on kyseessä toimintatila, jossa yhteys toiseen päätelaitteeseen ei ole vielä muodostunut. Toteutetussa ohjelmakoodissahan yhteyden avausyritys oli ajoitettu tapahtumaan 15 sekunnin välein, joten moduulin haluttu toiminta tältä osin voidaan näin ollen todentaa kuvaa tarkastelemalla. Kuvassa tasojen lisäksi näkyvät varsinaiset virtapiikit sen sijaan aiheutuvat Bluetooth -moduulin *page_mode* -asetuksesta. Oletusarvoilla tämä asetus asettaa moduulin muille laitteille havaittavaksi 1,28 sekunnin jaksotuksella. Edellä olevan kuvan skaalaus ei anna mahdollisuutta kyseisen sekvenssin erottamiseksi. Kuitenkin mittauksia suoritettaessa asteikkoa skaalattiin riittävästi toiminnan todentamiseksi, vaikkei tieto sinällään ollut olennainen. Yhteyden muodostuessa virrankulutus vakiintuu huomattavasti kuten jäljempänä oleva kuva 8.5 osoittaa.



Kuva 8.5. Virrankulutus Bluetooth -yhteyden aikana.

Kuvassa on siis kyseessä tilanne, jossa yhteys toiseen päätelaitteeseen on saatu yhdistettyä ja informaatio kulkee laitteiden välillä datapakettien muodossa. Kuvan virtapiikit aiheutuvatkin tällä kertaa juuri mainittujen datapakettien lähetyshetkistä. Yhteenvetona suoritettujen mittausten perusteella voidaan muodostaa seuraava taulukko 8.1, johon on koottu keskeisimpien toimintotilojen virrankulutusarvot koko keskusyksikölle. Virta -arvot on laskettu oskilloskoopin jännitearvoista virranmittausvastuksen suuruuden ollessa riittävän tarkasti tiedossa.

Taulukko 8.1. Virrankulutus eri toimintotiloissa.

TILA	VIRTA
Yhteyden muodostus	
hetkellinen maksimivirta	61mA
keskimääräinen virta	39mA
hetkellinen minimivirta	27mA
Yhteys avattu	
hetkellinen maksimiarvo	59mA
keskimääräinen virta	47mA

Teoriassa mitatut virrankulutusarvot lupaavat keskusyksikölle jopa 18 tunnin käyttöaikaa yhdellä latauksella. Teoria kuitenkin harvoin vastaa käytäntöä, sillä monet ulkoiset tekijät kuten järjestelmän käyttölämpötila vaikuttavat toiminta -aikaan. Esimerkiksi kylmissä olosuhteissa akun hyödynnettävissä oleva kapasiteetti vähenee nopeasti lyhentäen näin käyttöaikaa. Lisäksi tulee muistaa, että keskusyksikön

tehonsyötön lisäksi akun tulee syöttää teho myös langalliseen anturiverkkoon. Suoritettujen testien perusteella järjestelmä pysyy toimintakykyisenä keskimäärin 10 - 11 tuntia huoneenlämpötilassa, jota voidaan pitää kohtuullisena toiminta -aikana.

9. YHTEENVETO

Projektin tavoite oli suunnitella ja toteuttaa järjestelmä, joka tukisi tiettyä osa -aluetta henkilökohtaisessa terveydenhoidossa. Toteutukseksi valittiin järjestelmä, joka tukee kotioiloissa tapahtuvaa henkilön kuntoutumista lonkkaleikkauksoperaation jälkeen. Varsinaisen tutkimuksen tavoitteena oli tutkia ja testata mahdollisia menetelmiä ja tekniikoita, joilla kyseinen järjestelmä ja erityisesti keskusyksikkö olisi järkevästi toteutettavissa. Jo alussa oli tiedossa että tämän järjestelmän olennaisin osa, keskusyksikkö tarjoaisi paljon suunnittelu- ja toteutushaasteista. Suuria kysymysmerkkejä olivat muun muassa keskusyksikön integraation onnistuminen muuhun järjestelmään sekä laskentatehon riittävyys.

Luotettavan raajan asentotiedon tulkitsemisen kannalta oli hyvinkin epäselvää miten yksinkertaisella anturisolmutyypillä tehtävästä oli mahdollista selvittää. Vaikka itse anturisolmujen toteuttajana oli asiaan perehtynyt projektiosapuoli, ei käytetyillä antureilla täysin päästy toivottuun lopputulokseen. Loppupäätelmänä anturisolmujen suhteen oli, että osa solmuista olisi tarvinnut kolmiakselisen magneettianturoinnin käytetyn 2 -akselisen ratkaisun sijaan. Vasta tämä olisi mahdollistanut riittävän kompensointi -informaation määrän asennon laskennassa.

Toinen merkittävä kompastuskivi projektin kannalta oli käytetty ANT - radioliikenneprotokolla. Pienitehoisen radiolinkin kantama hukkui suureksi osaksi nestepitoisen lihaskudoksen aiheuttamiin häviöihin. Varsinkin solmut, joiden tarkoitus oli linkittyä raajan puolelta toiselle toimivat erittäin epäluotettavasti. Vaikka tiedettiin radion olevan pienitehoinen, mutta kuitenkin suunniteltu ihmisen ympäristön läheiseen tiedonsiirtoon, tuli radiolinkin heikko toimivuus suurena yllätyksenä.

Projekti, jossa on useita tutkimustahoja asettaa aina myös haasteita osapuolien väliselle tiedonsiirrolle. HipGuard -projektinkin puitteissa voidaan jossain määrin kyseenalaistaa oliko tiedon jakaminen riittävää tahojen kesken. Kenties joitain ongelmakohtia olisi voitu havaita paremman tiedonsiirron ansiosta riittävän aikaisin ja ratkaisumallit olivat olleet suunniteltavissa uudestaan. Projektia voidaan kuitenkin pitää myös hyvänä oppina laajan projektinhallinnan kannalta.

Projektin puitteissa saatiin joka tapauksessa toteutettua prototyyppitasen toimiva järjestelmä ja eritoten keskusyksikkö, joka jakaa ja hallinnoi järjestelmään olennaisesti kuuluvia radioliikennetekniikoita. Tämän lisäksi oli selvää jo projektin loppuvaiheessa,

että ainakin yksi järjestelmän osa -alue, painoa mittaava pohjallinen, oli herättänyt riittävästi mielenkiintoa jatko -projektin aikaansaamiseksi. Näiden selkeästi konkreettisten saavutusten lisäksi työn eri osa -alueet antoivat huomattavan määrän tietoa muun muassa Tampereen Teknillisen Yliopiston Kankaanpään Yksikön perustutkimukseen.

LÄHTEET

- [1] Syväriinen, Katri. Purkaako teknologia terveystommin. T -lehden artikkeli. 3/2006. [viitattu 12.10.2008]
Saataavissa: www.t-lehti.fi/story.php?page%3Dstory4800
- [2] Korhonen, I., Pärkkä, J. and van Gils, M. Health Monitoring in the Home of the future. IEEE Engineering in Medicine and Biology Magazine. [verkkolehti]. 22(2003)3. sivut 66-73. [viitattu 4.4.2008]
Saataavissa: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1213628
- [3] A brief History of Wearable Computing [viitattu 18.9.2008]
Saataavissa: <http://www.media.mit.edu/wearables/lizzy/timeline.html>
- [4] The Vision ja Reality of wearable Computing [viitattu 20.9.2008]
Saataavissa: www.wearable.ethz.ch/vision.0.html
- [5] Axisa, F., Schmitt, M., Gehin, C., Delhomme, G., McAdams, E. and Dittmar, A. Flexible Technologies and Smart Clothing for Citizen Medicine, Home Healthcare, and Disease Prevention. September 2005. 12s.
- [6] The MDKeeper, Specs and Usage. Tadiran Spectralink. 2005. [viitattu 12.3.2008]
Saataavissa: www.tadspec.com/lifecare/
- [7] Paradiso, R., Loriga, G. and Taccini, N. Wearable Health Care System for Vital Signs Monitoring. Medicon Conference. 2004. 5s.
- [8] Iso-Ketola, P., Karinsalo, T., Myry, M., Halme, A., Salpavaara, T., Lekkala, J. and Vanhala, J. Development of a lower extremity rehabilitation aid utilizing an insole-integrated load sensor matrix and a sole-embedded measurement node. ISWC'09. 13th IEEE International Symposium on Wearable Computers, 4 – 7 September, 2009, Linz, Austria. sivut 107-113.
- [9] Iso-Ketola, P., Karinsalo, T. and Vanhala, J. HipGuard: A wearable measurement system for patients recovering from a hip operation. PervasiveHealth 2008, 2nd International Conference on Pervasive Computing Technologies for Healthcare 2008. Tampere, Finland, Jan 29 – Feb 01 2008. 4s.
- [10] Myry, Manu. Liikkeentunnistus magneettiantureilla. Diplomityö. Tampereen Teknillinen Yliopisto. 2005. 78s.

- [11] KVH Industries Inc. Overview of Compass Technology. 15s. [viitattu 9.4.2009]
Saataavissa: <http://www.kvh.com/pdf/compassstech.pdf>
- [12] Webster, John G. The measurement, instrumentation and sensors handbook, 1st Edition. Chapter 17. CRCnetBase 1999. CRC Press LLC 1999. 2608s.
- [13] Foxlin, E. "Motion Tracking Technologies and Requirements," Handbook of Virtual Environment Technologies, chapter 8, K. Stanney, ed., Lawrence Erlbaum Publishers, Hillsdale, N.J., 2002, sivut 163-210.
- [14] Giorgino, T., Tormene, P., Lorussi, F., De Rossi, D. and Quaglini, S. Sensor Evaluation for wearable strain gauges in Neurological Rehabilitation. August 2009. 7s.
- [15] Mild, Pekka. Mat-2.129 Systeemien identifiointi, Kalman -suodin. Kevät 2003. 8s.
- [16] Salpavaara, Timo. Käyttäjän mittaaminen. Tampereen Teknillinen Yliopisto, sisäinen julkaisu. 2007. 3s
- [17] Atmel. Atmega644P, 8 -bit AVR Microcontroller with 64kbytes In-System Programmable Flash, Atmel Datasheet, 2008. 581s. [viitattu 5.4.2009]
saataavissa:
http://www.atmel.com/dyn/products/product_card_mcu.asp?part_id=4604
- [18] Nordic Semiconductor. nRF24AP1, Single chip 2,4GHz Transceiver with Embedded ANT Protocol. Nordic Semiconductor datasheet, 2009. 25s. [viitattu 6.4.2009]
saataavissa:
http://www.nordicsemi.com/files/Product/data_sheet/Product_Specification_nRF24AP1_1.0.pdf
- [19] Bluegiga Technologies. Bluegiga WT12, Bluetooth 2.0+EDR module, version 2.8. Bluegiga Technologies datasheet, 2009. 56s. [viitattu 6.4.2009]
saataavissa: http://www.bluegiga.com/more_wt12
- [20] Most important EMC design guidelines. [viitattu 29.10.2008]
Saataavissa:
www.cvel.clemson.edu/emc/tutorials/guidelines/Important_Guidelines.html

- [21] Jussi Mikkonen. Luotettava elektroniikka. Tampereen Teknillinen Yliopisto. 36s. [viitattu 12.8.2008]
- [22] Fjelstad, J. 2006. Flexible Circuit Technology, Third Edition. BR Publishing, Orlando, USA. 214s.
- [23] Hahto, Lasse. Taipuisan elektroniikan suunnittelu. Diplomityö. Tampereen Teknillinen Yliopisto. 2008. 93s.
- [24] Flexible Printed Circuit - Design Guide, koko manuaali [viitattu 11.6.2008]
Saataavissa: www.allflexinc.com/guid.shtml
- [25] Lenkkeri, J., Marjamaa, T., Jaakola, T., Karppinen, M. ja Kololuoma, T. Tulevaisuuden elektroniikan pakkaus- ja komponenttitekniikat. VTT Information Service, syyskuu 2003, sivut 28–39. [viitattu 17.10.2009]
Saataavissa: <http://www.vtt.fi/inf/pdf/tiedotteet/2003/T2213.pdf>
- [26] Puhvi -anturiverkon käyttämä protokolla. Tampereen Teknillinen Yliopisto, Tampere 2007, Julkaisematon sisäinen selvitys. 5s.
- [27] Bluegiga Technologies. iWRAP_2.2.0_User_Guide, koko manuaali, 4.9.2007 [viitattu 14.1.2008]
Saataavissa: <http://www.bluegiga.com/support>
- [28] Dvorak, Joseph L. Moving wearables into the mainstream – Taming the Borg. chapters 2 and 4. Springer Science+Business Media, 2008. 359s.
- [29] Bardoliwalla, D.F. Fast Curing, Low Exotherm Epoxy Potting and Encapsulating Systems. September 1997. sivut 245-247. [viitattu 11.11.2008]
saataavilla: ieeexplore.ieee.org/Xplore/dynhome.jsp
- [30] Santavirta, Sanni. Taipuisan elektroniikan suojaus elastomeerimateriaalilla. Diplomityö. Tampereen Teknillinen Yliopisto. 2007. 77s.
- [31] Kaappa, Emma. Tekstiilien liitos- ja leikkuuteknologiat elektroniikan suojaamisessa. Diplomityö. Tampereen Teknillinen Yliopisto. 2008. 80s.
- [32] Heikkinen, Korhonen, Manninen. Sulautettujen ohjelmistojen testaus. Prosessori maaliskuu/1998. 3s. [viitattu 12.12.2009]
Saataavissa: <http://www.prosessori.fi/es98/testaus.htm>

- [33] Tasse, Gregory. The Economy Impacts of Inadequate Infrastructure of Software testing. National Institute of Standards and Technology. May, 2002. 309s. [viitattu 4.1.2010]
Saatavissa: <http://www.nist.gov/director/prog-ofc/report02-3.pdf>
- [34] Pyykkö, Timo. Ohjelmistotestaus siirryttäessä perinteisestä ohjelmistokehitysmenetelmistä Scrumiin, Jyväskylän yliopisto, 2010. 166s.
- [35] Haikala I. & Märijärvi J. Ohjelmistotuotanto, 11. uudistettu painos. Jyväskylä: Gummerus kirjapaino. sivut 36-37.
- [36] Lammi, Hanna-Kaisa. Ohjelmistotuotanto – testaus. Opintomateriaali, luento 8. Lappeenrannan Teknillinen Yliopisto. 42s. [viitattu 10.1.2010]
saatavissa: www2.it.lut.fi/kurssit/02-03/010758000/luennot/luento8.ppt
- [37] Boehm B.W. Guidelines of Verifying and Validating Software Requirements and Design Specifications. Teoksessa: Samet P.A. Proceedings of Euro IFIP '79. North Holland 1979. sivut 711-719.

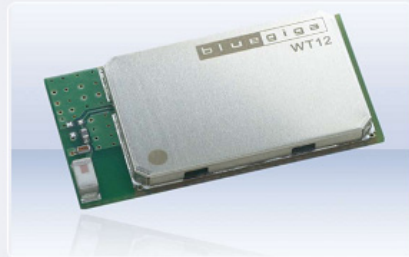
LIITTEET

Liite 1. Komponenttien datalehdet

14/10/09



WT12 Bluetooth® Module™ Description



WT12 is a next-generation, class 2, *Bluetooth* 2.1 + EDR module. It introduces three times faster data rates compared to the existing *Bluetooth* 1.2 modules even with a lower power consumption. WT12 is a highly integrated and sophisticated *Bluetooth* module, containing all the necessary elements from *Bluetooth* radio antenna to a fully implemented protocol stack. Therefore WT12 provides an ideal solution for developers who want to integrate *Bluetooth* wireless technology into their designs with limited knowledge of *Bluetooth* and RF technologies.

Key Features

- *Bluetooth* Class 2
- Integrated chip antenna
- Enhanced Data Rates (EDR) with data throughput up to 2-3Mbps
- Support for Adaptive Frequency Hopping (AFH) and 802.11 co-existence
- USB version 2.0
- UART with bypass mode
- 8Mbits of flash memory
- Supported *Bluetooth* profiles: SPP, DUN, OBEX OPP, HFP v.1.5, DID, HID + HCI
- Industrial temperature range from -40°C to +85°C
- RoHS compliant
- Pin-to-pin compatible with WT11 module
- Simple iWRAP™ firmware for controlling *Bluetooth* wireless technology
- Fully qualified end product with *Bluetooth* 2.1 + EDR, CE and FCC

WT12 module combined with Bluegiga's complete development, testing and verification services and excellent developer support, OEMs and designers ensure that their products reach the market rapidly and cost-efficiently in relation to time and resources. Bluegiga has extensive in-house knowledge of both software and hardware offering customers a single point of contact to all *Bluetooth* related issues.

By default WT12 module is equipped with powerful and easy-to-use iWRAP firmware. iWRAP enables users to access *Bluetooth* functionality with simple ASCII commands delivered to the module over serial interface. Entering the world of *Bluetooth* wireless technology could not be easier!

With iWRAP software you have several implementation options:

- ✦ iWRAP can be configured to operate autonomously, - as a *Bluetooth* cable replacer
- ✦ To create sophisticated applications - a host system can be used to control iWRAP with ASCII commands
- ✦ With The GPIO interface in WT12 module can be used to connect host and iWRAP

Besides the iWRAP firmware Bluegiga also offers several other firmware options for WT12 module. Standard Host Command Interface (HCI) firmware is supported and an ideal solution for systems where the host system is capable of running the entire *Bluetooth* stack and profiles and WT12 is utilized as the physical radio over UART or USB interface.



Information subject to changes



TECHNICAL DATA

Firmware

- iWRAP™ command interface to access the *Bluetooth* functionality and to configure the parameters with simple ASCII commands
- HCI firmware available for UART and USB interfaces
- Possibility to use / develop custom firmware
- 128-bit *Bluetooth* encryption available for all firmware options

Hardware

- *Bluetooth* Class 2 radio (range up to 30 meters)
 - Nominal output power +4 dBm
 - Nominal sensitivity -84 dBm
 - Uses 2.4 GHz ISM band
 - Based on CSR's BC04 chipset
- Integrated antenna
- Host processor interface with UART or USB
- SPI interface for firmware and parameter upgrades
- 6xGPIO
- PCM interface for audio applications
- Supply voltage: regulated 3.2 – 3.4 VDC
- Power consumption with iWRAP interface:
 - Link active min: 7mA, max: 60mA
 - Link active in sniff mode: 2.5mA
 - Link active in park mode: 2.5mA
 - Idle with deep sleep: <1mA
- PCB form factor: 26 x 14 x 2.3 mm
- Operating temperature: -40 °C to +85 °C
- Metal shielding to prevent RF interference
- Reference designs available for HCI, cable replacement and audio applications

Product Codes

- | | |
|------------------------------------|------------|
| • iWRAP Firmware, chip antenna | WT12-A-AI |
| • HCI Firmware (USB), chip antenna | WT12-A-HCI |
| • Custom Firmware, chip antenna | WT12-A-C |

Development and Evaluation

- WT12-A Evaluation Kit EKWT12-A
- CSR's BlueLab Professional SDK

Other Products

- WT11 Class 1 *Bluetooth* Module™
- WT32 *Bluetooth* Audio Module™
- Bluegiga Access Server™ 2291, 2293

Certifications

- *Bluetooth* 2.1 + EDR
- CE, FCC, IC

Applications

- Cable replacement
- Point-of-sales systems
- Barcode readers and pay terminals
- Telemetry and machine-to-machine devices
- Logistics and transportation systems
- Automotive inspection and measurement systems
- Medical systems
- Fitness and sports telemetry devices
- PDAs and other portable terminals
- PCs and laptops



For more information about Bluegiga Technologies, please contact:

Bluegiga Technologies Oy
 Sinikalliontie 5 A
 02630 Espoo, Finland
 Phone: +358 9 435 50 60
 Fax: +358 9 435 50 660
 www.bluegiga.com
 sales@bluegiga.com

© Bluegiga Technologies 2000-2009.
 Bluegiga Technologies takes no responsibility for any mistakes that might appear in this document. It reserves the right to change devices, software or specifications detailed here at any time without notice, and does not make any commitment to update the information contained here. Bluegiga products are not authorised for any use as critical components in life support devices or systems. Bluegiga Access Server, Access Point, iWRAP and WRAP THOR are trademarks of Bluegiga Technologies.
 The *Bluetooth* trademark and logo are registered trademarks and are owned by the *Bluetooth* SIG, Inc.

PRELIMINARY PRODUCT SPECIFICATION



Single chip 2.4 GHz Transceiver with Embedded ANT protocol

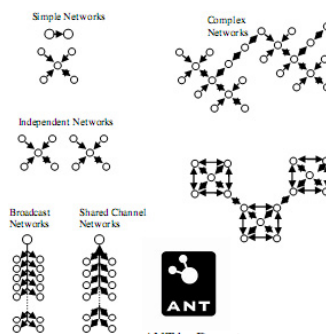
nRF24AP1

FEATURES

- Integrated PAN (Personal Area Network)
- Drop in wireless networking with simple serial interface
- 2.4GHz Worldwide ISM Band
- Ultra-low power (coin cell battery)
- Fully scalable
- Broadcast, Acknowledged or Burst Data
- Message rates 0.5Hz -> 200Hz (8byte data payload)
- Burst transfer rates up to 20kbps (true data throughput)
- Public and private networks
- 1 Mbps RF data rate
- 125 RF channels

APPLICATIONS

- Sensor Networks
- Industrial Automation
- Home Automation
- Sports Monitoring



ANT by Dynastream
© 2005 all rights reserved

GENERAL DESCRIPTION

The nRF24AP1 is an ultra-low power single-chip radio transceiver with embedded ANT protocol for personal area networks. The transceiver's RF operating frequency range falls within the world-wide 2.4 - 2.5 GHz RF ISM band, allowing for regulatory compliance and product sales into global markets.

QUICK REFERENCE DATA

Parameter	Value	Unit
Minimum supply voltage	1.9	V
Maximum output power	0	dBm
Maximum data rate (over the air)	1000	kbps
Temperature range	-40 to +85	°C
Sensitivity	-80	dBm
Average current consumption as low as	30	µA
Peak current consumption TX @ -5dBm	13.5mA for 350µs	mA
Peak current consumption RX	22mA for 600µs	mA
Max # of simultaneous connections ¹	>65000	connections
Max # of simultaneous independent 2-way connections	4	2-way connections
Maximum sustained transfer rate (all data – no overhead) ²	20	kbps
CR2032 Battery life in a typical sensor application ³	5	years

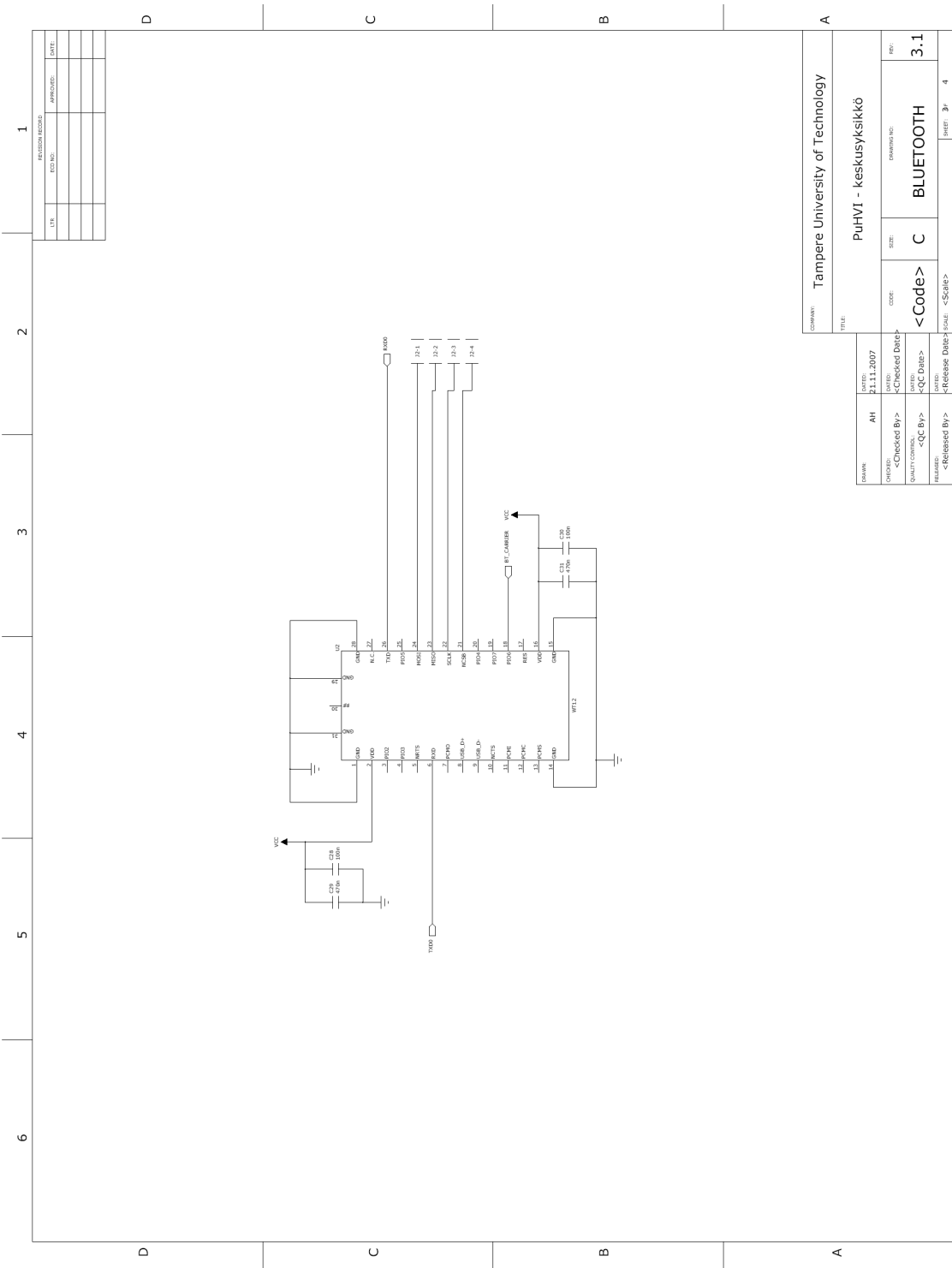
Table 1: nRF24AP1 quick reference data

¹ Using Shared Channel Network

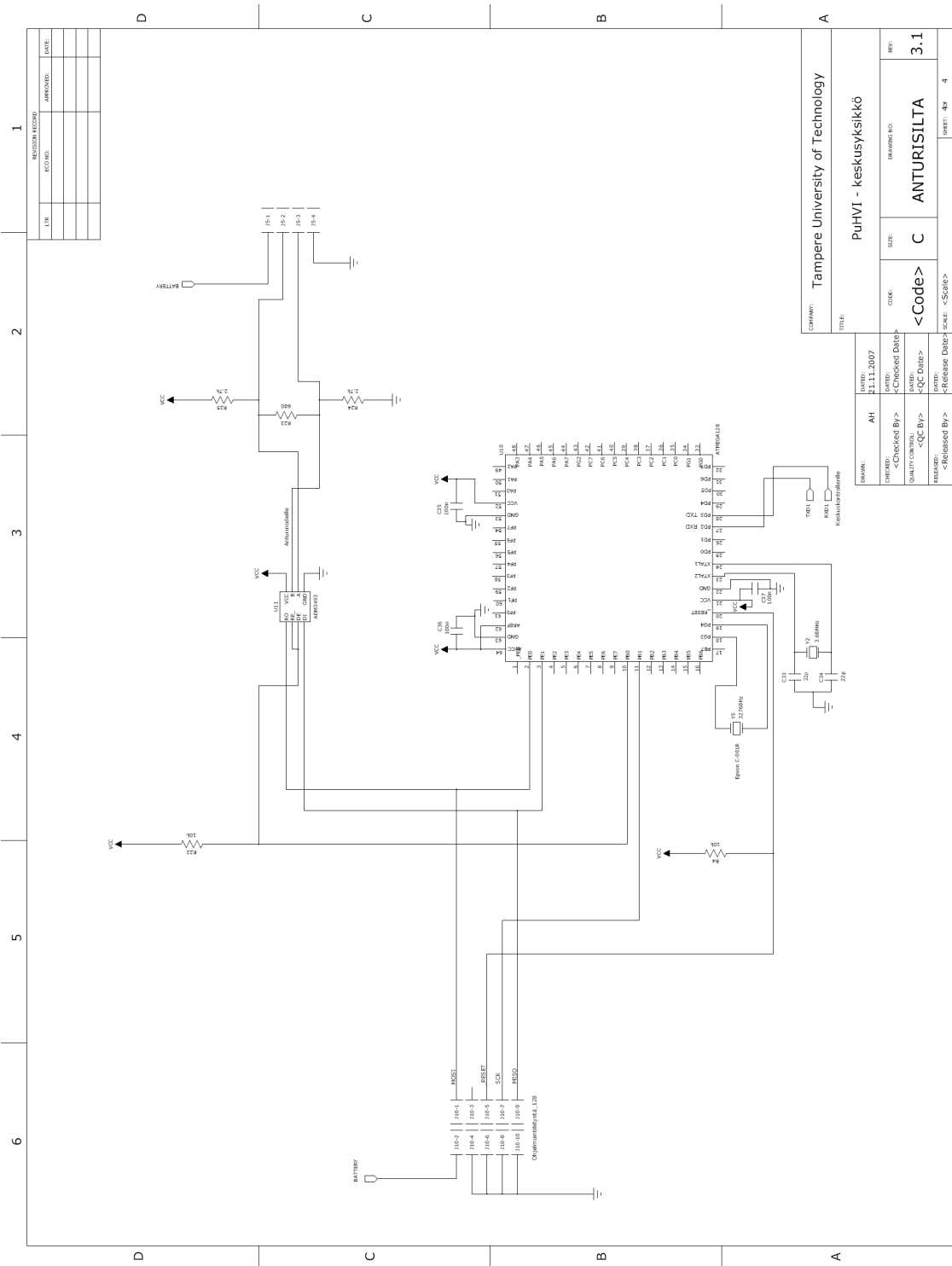
² Transfer rate refers to data rate of the end application's message payload

³ Message interval of 2s, 1 hour/day usage

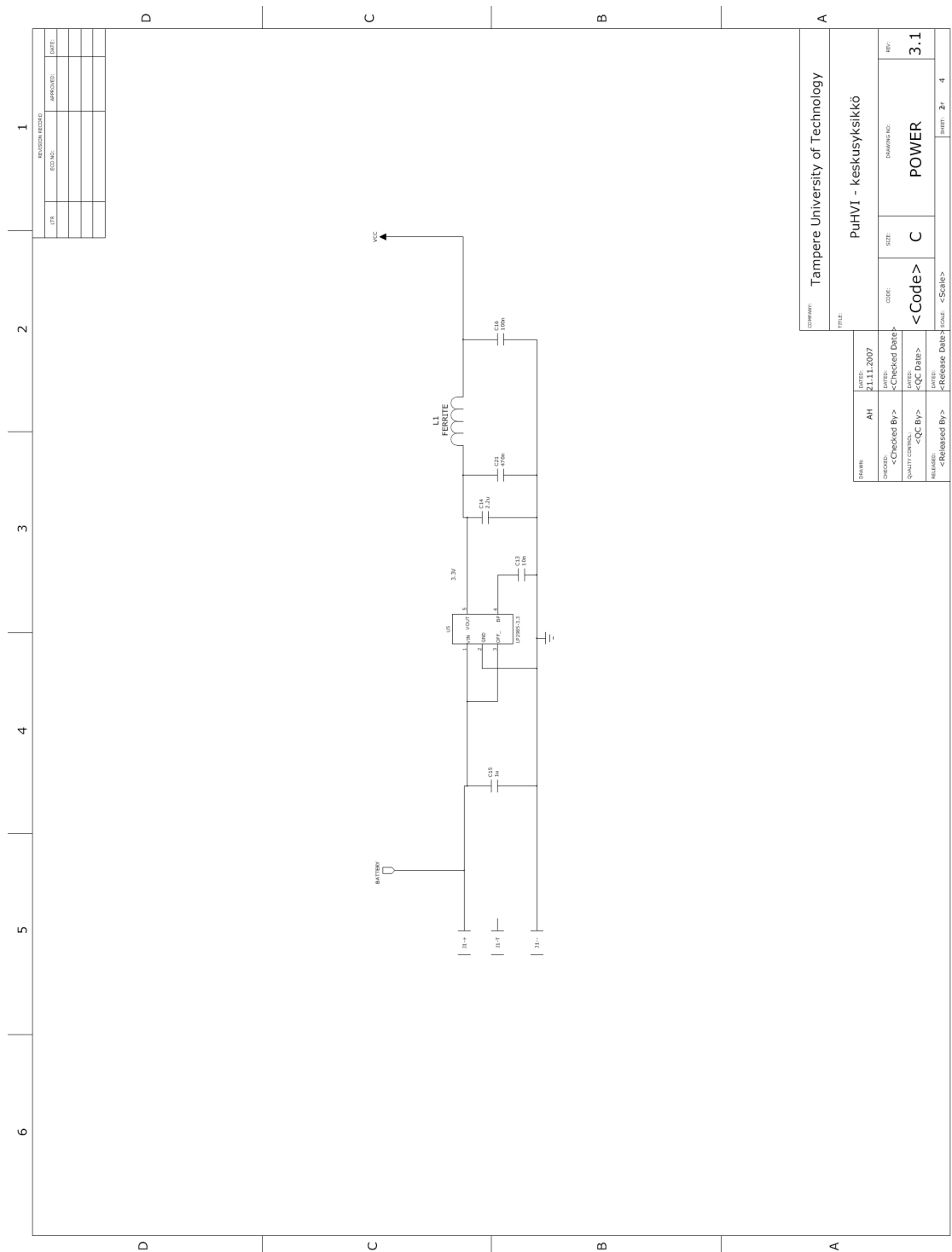
Piirikaaviot jatkoa



Piirikaaviot jatkoa



Piirikaaviot jatkoa



Liite 3. Olennaisia osia ohjelmakoodista

```

/*****
Anturiverkolta tulevia datapaketteja parseroiva
funktio HipGuard - ohjelmistoon.
Aki Halme, 2008
Tampereen Teknillinen Yliopisto
*****/
struct paketti receiveSensors() {
    unsigned char lenght = 0;
    struct paketti temp_packet;

    unsigned char low_X; //tarkisteet
    unsigned char low_Y;
    unsigned char low_Z;

    int check = 0;
    unsigned char calc_check = 0;
    unsigned char rec_check = 0;

    lenght = UART1_getc(); //paketin pituus ilman LEN-tavua talteen

    //vaaditaan 3 preamble-tavua
    int n = 0;
    while(n <= 2) {
        received_char = UART1_getc();
        if(received_char != 0xFF) {
            temp_packet.type = 0x31; //sync tavua ei saatu
            return temp_packet;
        }
        check = check + received_char;
        n++;
    }

    //osoite, josta vastaanotetaan
    temp_packet.addr = UART1_getc();
    check = check + temp_packet.addr; //tarkistussumman kokoaminen

    //paketin tyyppi
    temp_packet.type = UART1_getc();
    check = check + temp_packet.type; //tarkisteeseen

    //luetaan varsinainen dataosio vain jos vastaanotettu paketti on
    //tyypiltään _DATAREPLY(0x38)
    if(temp_packet.type == 0x38) {
        //datapaketin sisällän luku riippuu siitä mitä antureita on kytketty
        temp_packet.devices = UART1_getc();

        if(temp_packet.devices == 0x01) { //ainoastaan kiihtyvyysanturi
            //luetaan 3 kpl 16-bit etumerkitöntä kokonaislukua muuttujiin
            //muistetaan kasvattaa tarkistussummaa
            temp_packet.SCA_X = UART1_getc(); //ensin LSB
            check = check + temp_packet.SCA_X;
            temp_packet.SCA_X += (UART1_getc() << 8);
            check = check + temp_packet.SCA_X;

            temp_packet.SCA_Y = UART1_getc();
            check = check + temp_packet.SCA_Y;
            temp_packet.SCA_Y += (UART1_getc() << 8);
            check = check + temp_packet.SCA_Y;

            temp_packet.SCA_Z = UART1_getc();
            check = check + temp_packet.SCA_Z;
            temp_packet.SCA_Z += (UART1_getc() << 8);
            check = check + temp_packet.SCA_Z;

            sbi(PORTA, PA4); //led on
        }
        else if(temp_packet.devices == 0x05) {
            //luetaan 3 16-bit etumerkitöntä kiihtyvyysarvoa sekä
            //2 16-bit etumerkitöntä kompassiarvoa
            //muistetaan kasvattaa tarkistussummaa
            temp_packet.SCA_X = UART1_getc(); //ensin LSB
            temp_packet.SCA_X += (UART1_getc() << 8);

```

```

        check = check + temp_packet.SCA_X;

        temp_packet.SCA_Y = UART1_getc();
        temp_packet.SCA_Y += (UART1_getc() << 8);
        check = check + temp_packet.SCA_Y;

        temp_packet.SCA_Z = UART1_getc();
        temp_packet.SCA_Z += (UART1_getc() << 8);
        check = check + temp_packet.SCA_Z;

        temp_packet.MS_X = UART1_getc();
        temp_packet.MS_X += (UART1_getc() << 8);
        check = check + temp_packet.MS_X;

        temp_packet.MS_Y = UART1_getc();
        temp_packet.MS_Y += (UART1_getc() << 8);
        check = check + temp_packet.MS_X;
    }
    //jos on positiivinen kuittaus
    if(temp_packet.type == 0x30) {
        return temp_packet;
    }
    //luetaan vielä tarkistussumma
    rec_check = UART1_getc();

    //tarkistussumman laskeminen
    calc_check = check;
    calc_check ^= 0xFF; //summa 8-bittiseksi

    //jos tarkiste täsmäsi
    if(calc_check == rec_check) {
        return temp_packet;
    }
    return temp_packet;
    //väärä tarkiste lähetetään negatiivisena kuittauksena
    temp_packet.type = 0x31; //tätä tutkitaan ensimmäisenä lähetetystä paketista
    return temp_packet;
}

/*****
Bluetooth -moduulin WT12 alustusfunktio, josta nähtävillä
moduulin asetusarvot.
Aki Halme, 2008
Tampereen Teknillinen Yliopisto, Elektroniikan laitos
*****/

int initBT(void) {
    init_UART0();
    //poistetaan laiteparit muistista
    send_BT("SET BT PAIR *\r\n");
    //uusi laitepari
    send_BT("SET BT PAIR 00:16:38:C0:DB:3E 18594abcd4859ef6a37b8d354dac183f\r\n");
    send_BT("SET BT NAME Puhvi\r\n"); //moduulin nimi
    send_BT("SET BT CLASS 001f00\r\n");
    send_BT("SET BT AUTH * 1234\r\n"); //asettaa pin-koodin
    //keskusyksikkö näkyvillä ja vastaa kutsuihin kunnes yhteys muodostetaan
    send_BT("SET BT PAGEMODE 4 3000 1\r\n");
    send_BT("SET BT ROLE 0 f 7d00\r\n");
    send_BT("SET CONTROL BAUD 115200,8n1\r\n"); //UART-parametrien asetus
    send_BT("SET CONTROL CD 00 0\r\n");
    send_BT("SET CONTROL ECHO 0\r\n"); //moduulin kaiutus-mode
    send_BT("SET CONTROL ESCAPE 43 00 1\r\n");
    //yritetään yhdistää paritettuun laitteeseen 15s välein
    //send_BT("SET CONTROL AUTOCALL 1101 15000\r\n"); //yhteys PC:lle
    send_BT("SET CONTROL AUTOCALL 10ff 15000\r\n"); //puhelimelle 10ff

    return 1;
}

```